Title

Researcher reasoning meets computational capacity: Machine learning for social science

Authors

Lundberg, Ian

Brand, Jennie E

Jeon, Nanum

Publication Date

2022-11-01

DOI

10.1016/j.ssresearch.2022.102807

Peer reviewed

Human reasoning meets computational power: Machine learning for social science [*]

Ian Lundberg[†] Jennie E. Brand[‡] Nanum Jeon[§]

February 18, 2022

Word count: 9,722

Abstract

Computational power and digital data create new opportunities to explore and understand the social world. A special synergy is possible when social scientists combine human attention to certain aspects of the problem with the power of algorithms to automate other aspects of the problem. We review selected exemplary applications where machine learning amplifies human coding, targets human attention, and relaxes certain assumptions. We then seek to reduce perceived barriers to machine learning by summarizing several fundamental building blocks and their grounding in classical statistics. We close by presenting a few guiding principles and promising approaches where we see particular potential for machine learning to transform social science inquiry. Our aim is to convince social scientists that machine learning tools are accessible, worthy of attention, and ready to yield new discoveries.

Some guiding questions for lab:

1. What part do you like?
2. What part is confusing or needs work?
3. Is there text that would benefit from a figure?
4. All reactions and thoughts are welcome!

[*] Draft in progress. Please do not cite or ciculate. Research reported in this publication was supported by the National Science Foundation under Award Number 2104607 and by the Eunice Kennedy Shriver National Institute of Child Health & Human Development of the National Institutes of Health under Award Number P2CHD041022.

[†] Department of Sociology and California Center for Population Research, UCLA, ianlundberg.org, ianlundberg@ucla.edu.

[‡] Department of Sociology, Department of Statistics, and California Center for Population Research, UCLA, profjenniebrand.com, brand@soc.ucla.edu.

[§] Department of Sociology, Department of Statistics, and California Center for Population Research, UCLA, https://soc.ucla.edu/grads/nanum-jeon, njeon@ucla.edu

Contents

1 Introduction

Advances in statistics and machine learning have the potential to rapidly expand the toolkit available to social scientists. The pace of change will depend on how social scientists weigh the costs and benefits of adopting new tools. Our review of existing work emphasizes three key benefits: machine learning can amplify human coding, target human attention, and relax certain assumptions. These benefits unlock new research directions, from the study of digital datasets of previously intractable size to the estimation of causal effects with minimal functional form assumptions. Yet many social scientists have yet to adopt machine learning tools despite their promise. One reason machine learning methods have appeared infrequently thus far may be the appearance of high adoption costs, such as the time needed to learn new methods and the difficulties that arise when interpreting a complex model. Yet the increasing availability of open-source software and pedagogical materials means that these costs are constantly falling. A uniting theme of our review is an argument that the benefits of machine learning are likely to substantially outweight the costs over time.

Related to assumed costs, social scientists may have a preconception that the adoption of machine learning methods requires a qualitative shift away from classical statistical methods. A second theme of our review is that there is no such qualitative shift. Where possible, we trace the lineage of machine learning methods directly back to standard statistical tools. While the fields of "statistics" and "machine learning" have at times differed in their emphasis on various aspects of data analysis (Breiman, 2001), many of the key advances occur when these perspectives are brought together (e.g., Wager and Athey 2018). What unites these fields is far greater than what divides them. When a social scientist uses a statistical method, they can conceptualize that method as a specific case of a machine learning tool. The question is not whether to adopt machine learning—every quantitative social scientist has already done that by using any statistical method. The question is whether to broaden the scope of machine learning tools in your toolkit. To the degree that researchers say yes to that question, the pace of change in social science may be rapid.

Our argument proceeds in several sections. We first emphasize several benefits of machine learning by reviewing its use in existing social science research. Second, we provide a pedagogical introduction to some of the central building blocks of machine learning, with a special emphasis on their connection to standard statistical approaches. Third, we discuss some frontiers of machine learning research which may be especially fruitful for social science research in the future. Finally, we conclude with a discussion of how machine learning may contribute to social science knowledge moving forward.

2 What you can do with machine learning: Exemplary applications in social science

Machine learning is already yielding new insights within social science. To uncover exemplary applications, we reviewed papers published from 2016–2021 in a set of journals drawn from sociology (*American Sociological Review* and *American Journal of Sociology*),[1] political science (*American Political Science Review*), and economics (*American Economic Review*). We also reviewed articles appearing in one interdisciplinary journal (*Social Science Research*) and included a few articles from methodological journals (*Sociological Methodology* and *Political Analysis*). We do not review all uses of machine learning, nor do we review all classes of machine learning methods. Instead, we highlight exemplary cases that illustrate three high-level ways that machine learning can transform the research process: machine learning can amplify human coding, target human attention, and relaxes certain assumptions. We also offer a word of caution: there are problems machine learning does not solve, and researchers must beware of these limitations. We illustrate this word of caution by discussing the use of prediction to guide policy interventions.

2.1 Machine learning can amplify human coding

One characteristic of the digital age is a high volume of available data in unstructured formats, such as text, audio, and video. Social scientists might like to convert these documents, sound bites, and video clips into a small set of categories relevant to a research question. That type of labeling requires human attention. It is feasible at a small sample size. Yet human coding is prohibitive for the massive number of cases available in digital datasets. A promising use of machine learning is to amplify human coding: if the researcher manually labels a random sample, automated methods can learn patterns in that sample and predict in the much larger population.

[1] We considered two sociology journals because each sociology journal contained fewer applications of machine learning than in political science or economics.

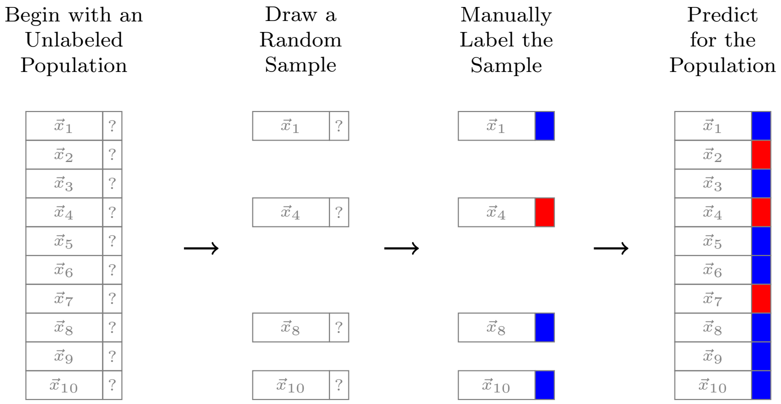


Fig. 1. Machine learning can amplify human coding. One setting which is particularly promising for machine learning exists when social scientists have many observations, each of which contains some high-dimensional predictor set $\vec{x}_i$ (e.g., the text of document i) but the researcher is interested in some low-dimensional, unobserved categorization y_i (e.g., the topic of document i, here represented by colors). A researcher who manually codes a random sample of the observations into categories can use machine learning tools to amplify that coding by predicting for the full population.

For example, King et al. (2017) studied government involvement in the social media ecosystem in China. In one analysis, they examined 43,757 social posts made by individuals employed by the Chinese government to spread propaganda. This volume of digital data would be extremely costly to analyze by hand. Instead, the authors drew a random sample of 200 posts and hand-coded them into a set of categories chosen by the authors. Using these 200 posts, they learned the statistical patterns linking the words used in the posts (predictors) to the categories defined by the researchers (labels). Finally, they used these patterns to estimate the prevalence of each category of post in the entire set of 43,757 posts (using a pre-established procedure available in open-source R software; see Hopkins and King 2010 and Jerzak et al. 2019). The authors were then able to show that roughly 80% of the posts did not engage in arguments about the Communist Party but instead simply involved cheerleading for China and for the Party. This descriptive evidence was made possible by human expertise (defining the categories of posts and labeling a sample) amplified by the power of machine learning (to draw inference about a massive data set).

Amplification of human coding applies to many questions involving high volumes of text data. Similar to the study above, Su and Meng (2016) manually categorized the topics of 1,000 messages from citizens to Chinese provincial officials and then used supervised methods to make predictions for the topics in the full set of 207,554 messages. In an entirely different context, Friedman and

Reeves (2020) explored patterns of cultural distinction in recreational activities by studying the lives of 71,393 British elites over the 19th and 20th centuries who appeared in *Who's Who*, a book cataloguing their lives. They manually coded 600 entries into three categories of recreational activities—aristocratic, highbrow, or ordinary—and then used supervised learning to estimate the prevalence of each type of recreation in the full set of 71,393 entries. The authors then summarized how patterns of elite portrayal of their recreational activities changed over time, an exercise which was only possible by combining human decisions (categorizing text into these three categories) amplified by machine learning to draw inference in a massive sample.

Beyond text, new forms of audio and visual data also become amenable to analysis through a strategy of amplified human coding. Knox and Lucas (2021) note that political scientists often study political speech by first transforming an audio file into a transcript, thus discarding all of the audio information. In a sample of audio files from Supreme Court hearings, they manually label some speech patterns as demonstrating skepticism toward the presented argument, and then they develop methods to predict skepticism in unlabeled utterances as a function of the audio profile of those utterances. Images are likewise amenable to analysis by amplified coding, following well-established methods for computer vision (Szeliski, 2010). Using 53,249 images of vote tally sheets from a Mexican election in which fraud was suspected, Cantú (2019) labels a random sample of 900 images as containing alterations or no alterations. Cantú (2019) then learns a convolutional neural network classifier, evaluates on a validation set of 150 images, and predicts alterations or no alterations for the full 53,249 images to reveal the extent of fraud by election officials who modified the vote tallies.

Amplified human coding is powerful because it draws on the strengths of machine learning and social science. The social scientist takes a high-dimensional predictor $\vec{x}$ (text, audio, or video data) and converts that into a category y among a few discrete choices constructed for their relevance to the theoretical question. This step requires social science theory to define the categories. Then, a machine learns the underlying mapping and uses it to predict for a sample size which would be prohibitively large for direct human coding. Machine learning tools thus amplify a labeling task which is fundamentally human at its core.

2.2 Machine learning can target human attention

In contrast to amplified human coding, there exist other settings in which the researcher does not know what aspects of the data may be most interesting. To the degree that the researcher can formalize what would make an aspect of the data interesting, an algorithm can search through the data to target human attention.

For example, political scientists often use conjoint experiments (Hainmueller et al., 2014) to examine how voters respond to a variety of signals about a candidate. For example, Breitenstein (2019) presented voters with profiles of hypothetical mayoral candidates and randomly varied signals of the candidates' sex, party affiliation, experience qualities, economic performance under their leadership, and evidence of corruption. The number of treatment conditions is numerous, with $2 \times 4 \times 2 \times 2 \times 3 = 96$ unique profiles possible by combining these attributes. Breitenstein (2019) summarized the causal effects by a linear regression to estimate the average effect of each component, marginalized over all the other components. In a reanalysis, Incerti (2020) relaxes this additive approximation by using a decision tree—a machine learning tool to partition the population into subgroups as an interactive function of the predictors. Decision trees recursively split the data into subsets where outcomes are increasingly homogeneous. The discovered interactions in this setting are interesting. Voters were most likely to support a non-corrupt politician's profile (72% in support) but also demonstrated high support for profiles involving corruption as long as the candidate was of the same party as the respondent and had a history of good economic performance under their leadership (67% in support). Meanwhile, a corrupt candidate of a different political party from the respondent garnered only 36% support. This kind of interactive relationship—supporting a corrupt candidate if and only if they have certain other desired characteristics—would be hard to predict a priori. When presented with a large set of 96 treatment conditions, a machine learning tool (a decision tree) can guide attention toward treatment conditions with particularly interesting outcomes.

Machine learning can also target human attention in settings with one binary treatment variable. In these settings, one might search for population subgroups across which the effect of the binary treatment varies substantially. Athey and Imbens (2016) developed causal trees, an extension of decision trees specifically designed to uncover effect heterogeneity. In one application, Brand et al.

(2021) assessed variation in the effects of college completion on low-wage work. They found that college completion reduced low wage work most for individuals whose mothers had less than a high school degree, who grew up in large families, and who had low social control have the largest effects of college completion on reducing low-wage work. Not only does the use of causal trees allow researchers to uncover subgroups not previously considered, it also more transparently depicts the analyses that lead researchers to focus on particular subgroups. When a researcher chooses to highlight the outcomes of a particular subgroup, it is difficult to know how they came to that decision. When a causal tree highlights a particular subgroup, the algorithm that determines the highlighted result is fully transparent.[2]

2.3 Machine learning can relax certain assumptions

Conceptual argument often guides social scientists toward a particular research question and a particular set of variables to study. Yet conceptual argument often breaks down in the final step of the analysis: selection of a statistical model which may involve an assumed functional form linking the predictors to the outcome. Machine learning methods are especially helpful in this step: one can allow the data to select a statistical model using out-of-sample predictive performance as a criterion.

For example, Dube et al. (2020) use web scraping to gather data on Human Intelligence Tasks (HITs) posted online on Amazon Mechanical Turk (MTurk), which would include tasks like placing labels on images or completing a short questionnaire. Workers can see information about the available tasks including the financial reward offered for completion before deciding whether to complete that task. A HIT remains available until the poster has received their desired number of responses. The authors are particularly interested in the causal effect of the reward amount on the duration of time that the HIT remains posted, taken as a metric of how quickly workers sign up and complete the task. But there is a problem: whether workers choose to complete a task may also be a function of other aspects of that task, such as the title, keywords, and time allotted by

[2] Causal trees do not always discover effect heterogeneity. Sometimes, they reveal a surprising lack of effect heterogeneity. Handel and Kolstad (2017) analyzed a randomized health intervention and found almost no evidence of heterogeneity across the measured variables. Davis and Heller (2017) found that a randomized youth intervention in Chicago had roughly the same effect on arrests in all subpopulations studied. In general, a lack of evidence for effect heterogeneity does not mean that effects are constant for everyone, but only reveal a lack of evidence for heterogeneity as a function of the measured variables.

the requester. To proceed, the authors first need untestable identification assumptions for which machine learning is not helpful. In this settings, one must assume that the relationship between reward and duration is entirely causal within subgroups defined by the measured variables (Imbens and Rubin, 2015; Pearl, 2009). Given this assumption, the authors can identify the causal effect by adjusting for these variables. When deciding how to carry out that adjustment, theory is of limited use and machine learning may be quite helpful: machine learning can select a model to predict the treatment given the confounders or a model to predict the outcome given the treatment and confounders. Dube et al. (2020) use double machine learning (Chernozhukov et al., 2018) to adjust for confounding by learning an ensemble that averages over several learning algorithms to predict the treatment (reward amount) and the outcome (duration of posting). This machine learning strategy thus handles difficult statistical choices automatically, allowing the authors to focus their attention on the definition of the research question and the validity of the required causal assumptions.

As another example, researchers may seek to draw inference about a population using a non-representative sample. Gelman and Little (1997) proposed to accomplish this task by a parametric method: estimate a multilevel model for the survey responses as a function of measured variables (e.g., race, age), predict the outcome in each subgroup defined by those variables, and post-stratify by the known population distribution of the predictors. The validity of this procedure relies not only on an identification assumption (ignorable sample inclusion within strata of covariates), but also on the assumed functional form of the regression model. Bisbee (2019) relaxed the latter assumption with a nonparametric machine learning approach (Bayesian Additive Regression Trees). Which approach is superior will depend on the sample size and on how closely the standard parametric assumptions approximate the true response surface, which can be assessed by out-of-sample predictive performance. Overall, this example illustrates how nonparametric machine learning methods can directly estimate unknown functional forms while leaving all other aspects of the research question as they would have been under standard statistical procedures.

2.4 A word of caution: Machine learning, causal inference, and policy

To make the most of machine learning, social scientists must recognize what it can and cannot do. In particular, machine learning can describe the world as it exists but does not inform policy (what would happen under an intervention to change the world) in the absence of additional assumptions.

To illustrate this point, we consider tasks which Kleinberg et al. (2015) call "prediction policy problems." In these tasks, it may appear that public policy can be improved by incorporating predictions from machine learning algorithms. This is true only under certain assumptions about the causal effect of those interventions. A simplified yet illustrative example from Kleinberg et al. (2015) is the decision about whether to carry an umbrella. The causal effect is known a priori: if it rains, carrying an umbrella will cause you to remain dry. What is unknown is the risk of rain: it would be nice to have a machine learning algorithm to predict the probability of rain, so that we can target the intervention (carry the umbrella) on days when rain is likely.

Moving outside the idealized example and into real policy questions, however, the causal effect central to the claim is often much more complex. For example, Chalfin et al. (2016) consider whether firing some police officers and replacing them with other officers could reduce the rate of police shootings in Philadelphia. For this policy, the central question is causal: if we took a given encounter between a police officer and a civilian but counterfactually changed the officer involved, would the probability of a police shooting decrease? The question is difficult to answer. Some officers may shoot civilians at higher rates than other officers, but perhaps that is because of the tasks to which those officers are assigned (e.g., particularly dangerous neighborhoods). Perhaps any officer assigned to that task would shoot at the same rate. The authors term this problem "task confounding." To conclude that differences across officers are caused by differences in the officers rather than the tasks, the authors assume the absence of task confounding. Under this causal assumption, Chalfin et al. (2016) get to draw a causal conclusion: firing the 10% of officers with the highest propensity to shoot and replacing them with officers of average propensities to shoot would reduce shootings by 4.81 percent. Importantly, while the authors emphasize the use of predictive modeling, the conclusion rests critically on causal assumptions (task confounding). This case illustrates how the dichotomy between prediction policy problems and causal problems is false: a prediction problem that involves a policy intervention is fundamentally a causal problem, and it is dangerous to think that machine learning tools can solve these causal problems on their own. In general, prediction policy problems are only prediction problems under the assumption that the relevant causal effect is identified. Or better yet, the relevant causal effect may be known: we do not need an experiment to know that an umbrella will keep you dry if it rains. The more the causal effect is known, the more the researcher can focus on the predictive side of the problem. At the end of the next section, we discuss settings where causal effects are highly unknown and social scientists need to combine causal assumptions with predictive tools to answer policy questions.

3 Conceptual building blocks: The statistical foundations of machine learning

To realize the benefits discussed above does not require years of training in machine learning. Rather, researchers trained in classical statistics already possess some knowledge of the fundamental building blocks that support machine learning. This pedagogical section links machine learning to classical statistics by presenting a set of core concepts: task clarity, the bias-variance tradeoff, data-driven estimator selection, and tasks involving a new target population.

3.1 Task clarity: Define a precise goal

Every statistical problem begins with a task—the goal that we hope to accomplish. For instance, we might wish to make predictions in a particular setting or estimate a mean in some population. A precise statement of the task is essential in all quantitative research, and it takes on renewed importance in the context of machine learning, which can often be tailored specifically to the task at hand. For example, consider a task which has been well-studied in both statistics and machine learning: drawing inference about a target population from a sample. We discuss this task from two perspectives: estimation of unknown population parameters and prediction of out-of-sample cases.

Suppose a researcher studies academic performance for students nested within classrooms. Each student i has a test score Y_i capturing their academic performance. We would like to understand how test scores vary across classrooms,

$$\theta_j = \mathbf{E}(Y_i \mid J_i = j) \tag{1}$$

where $J_i = j$ means we are taking the expectation among students in classroom j. Equivalently, we can conceptualize θ_j as a prediction rule: if we see a new student in class j, we would predict

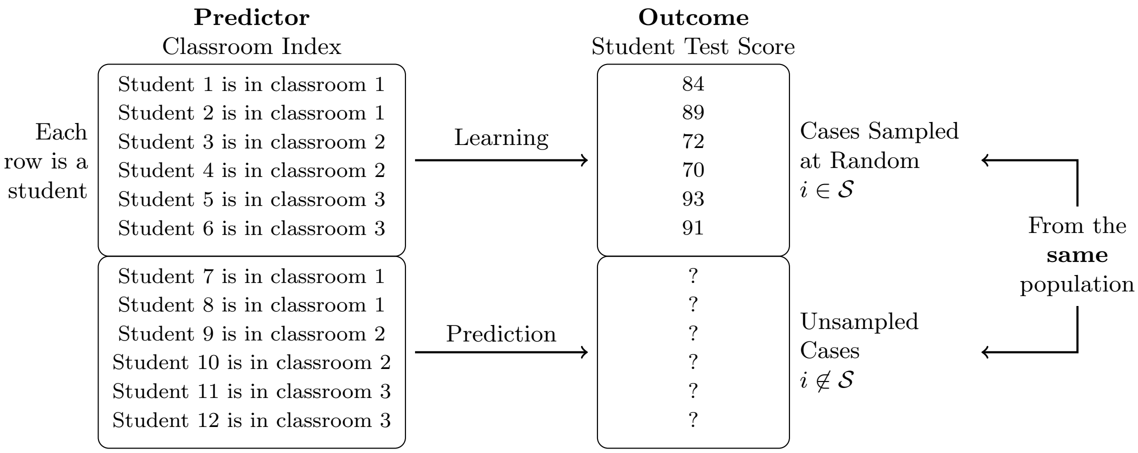

Predictor
Classroom Index
Outcome
Student Test Score
Each row is a student
Student 1 is in classroom 1
Student 2 is in classroom 1
Student 3 is in classroom 2
Student 4 is in classroom 2
Student 5 is in classroom 3
Student 6 is in classroom 3
Learning
84
89
72
70
93
91
Cases Sampled at Random $i \in \mathcal{S}$
From the **same** population
Student 7 is in classroom 1
Student 8 is in classroom 1
Student 9 is in classroom 2
Student 10 is in classroom 2
Student 11 is in classroom 3
Student 12 is in classroom 3
Prediction
?
?
?
?
?
?
Unsampled Cases $i \notin \mathcal{S}$

Fig. 2. Task clarity: Out-of-sample prediction. A well-studied machine learning task involves a random sample $\mathcal{S}$ taken from a target population, where the goal is to learn a prediction function to predict the outcomes of new samples from that same target population. For instance, we might use classroom indices to predict the tests scores of individual students who were not observed in the training sample.

that student's unknown test score to be θ_j (Fig 2).

If we observed all students in every classroom, we could calculate each θ_j directly by the classroom mean. If we only observe a random sample $\mathcal{S}$ of students, then we need an estimator for this unknown parameter. For instance, we could estimate by the sample mean,

$$\hat{\theta}_j^{\text{Mean}} = \bar{y}_j = \frac{1}{|\mathcal{S}_j|} \sum_{i \in \mathcal{S}_j} Y_i \tag{2}$$

where the term beneath the summation sign indicates that we are summing over all students i in the sample $\mathcal{S}_j$ from classroom j. The sample mean is a consistent and unbiased estimator, yet it may not be the optimal estimator in a finite sample. We discuss this issue in the next section.

Estimation of class-specific means is a useful example because it bears resemblance to both statistics and machine learning. Social scientists and statisticians could easily study this problem without conceptualizing it as a machine learning problem. Yet it also contains several hallmarks of machine learning. Machine learning estimators often involve a very large set of parameters to be estimated (e.g., many classes) and apply in settings where the sample size seems large (e.g., many students) but in fact are small given the large number of predictors (e.g., few students in each classroom).

3.2 The bias-variance tradeoff: Choose a biased estimator

Continuing the example of students in classrooms, suppose that the sample size $|S_j|$ in class j is small (e.g., 5 students). In this case, the sample mean may be a poor estimator of the population mean in the classroom because the sample size is so small. For every statistical and machine learning estimator, a first-order concern is how well we can expect that estimator to accomplish our task. We want to choose an estimator which will be close to the truth on average when applied to hypothetical sample from the population. Counterintuitively, to produce an estimator which is close to the truth on average one might be well-advised to choose an estimator which has low variance but is slightly wrong on average—a biased estimator. Many of the best statistical estimators and nearly every estimator that would be considered "machine learning" accepts some bias in order to improve performance. To make the most of machine learning, social scientists will need to come to appreciate the benefits that bias can bring. We illustrate this point through an example which is standard in statistics: a multilevel model.

To better estimate the classroom mean in a small sample, the researcher could add a shrinkage term to produce a multilevel model estimator (Bryk and Raudenbush, 1992),

$$\hat{\theta}_j^{\text{Multilevel}} = \bar{y}_j - \underbrace{\frac{\frac{1}{n_j}\hat{\sigma}_j^2}{\frac{1}{n_j}\hat{\sigma}_j^2 + \hat{\delta}^2}\left(\bar{y}_j - \bar{y}\right)}_{\substack{\text{Shrinkage Term} \\ \text{(creates bias)}}} \tag{3}$$

where $\hat{\sigma}_j^2$ is the empirical variance of test scores across students within class j, $\hat{\delta}^2$ is the empirical variance of classroom-level mean test scores across all classrooms, and $\bar{y}$ is the mean test score in the entire sample.[3] The estimator $\hat{\theta}_j^{\text{Multilevel}}$ is a partial pooling estimator because it pools information from class j together with other information about the mean test score in the sample overall. The consequence of partial pooling is that the estimator for each class is biased toward the overall mean—the greater the shrinkage, the more the bias. Yet shrinking toward the overall mean also yields the benefit of reduced variance. Fig 3 shows that the amount of shrinkage in $\hat{\theta}_j^{\text{Multilevel}}$ is the amount that mimimizes the expected squared error of the estimator: across repeated samples, the average squared distance between the estimated mean and the truth.

[3] This estimator is sometimes called the "Best Linear Unbiased Predictor" FIND PAGE, although that name is misleading because the estimator is biased.

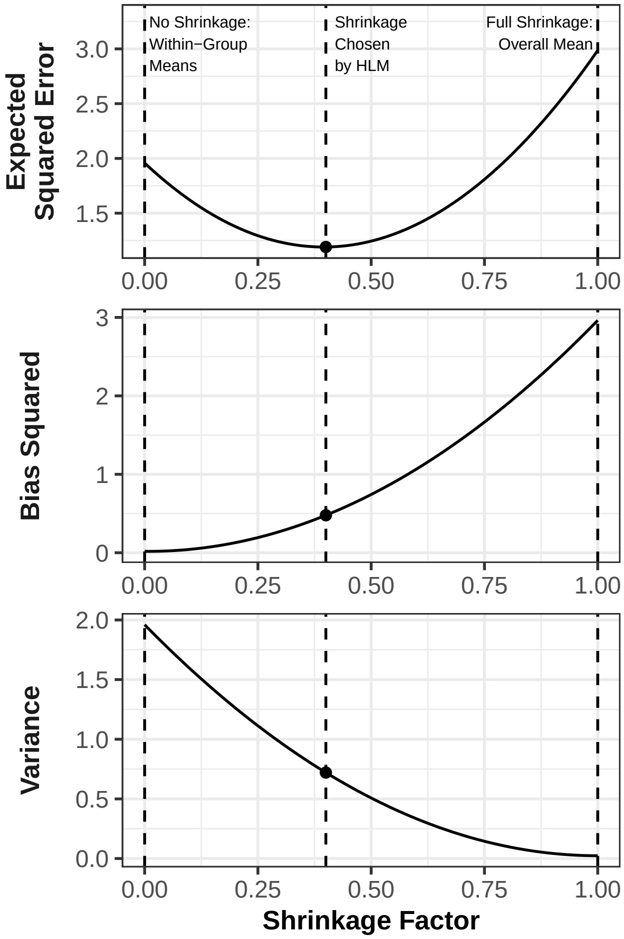


Fig. 3. Simulation: A hierarchical linear model balances the bias-variance tradeoff. In this simulation, there are 100 classes with class-level mean test scores normally distributed with variance 3. Within classes, student scores are normally distributed with variance 10. In each of 100 simulated samples, we estimate from a sample of 5 students from each class. The estimator partially pools the class-specific mean with the overall mean according to a shrinkage factor: $\hat{\theta}_j^{\texttt{(shrinkage factor)}} = \bar{y}_j - \texttt{(shrinkage factor)}\,(\bar{y}_j - \bar{y})$. A shrinkage factor of 0 involves no pooling so that the estimate is the sample mean within each class, and a shrinkage factor of 1 involves complete shrinkage so that the estimate for every class equals the overall sample mean. A hierarchical linear model selects a shrinkage factor equal to the variance of the within-class means divided by that variance plus the variance of the means across classes. The center dashed line takes those variances as known and shows that the multilevel shrinkage minimizes the expected squared error. To create each curve, we first calculate the statistic over simulations within classes, and then we report the average of the statistic over all classes.

The notion of accepting some optimal amount of bias in order to reduce the variance of an estimator is an idea that is much broader than multilevel models. In particular, the expected squared error of any estimator can be decomposed into components corresponding to bias and variance.

$$\text{Bias-Variance Tradeoff:} \qquad \underbrace{\mathbf{E}\left(\left(\hat{\theta}-\theta\right)^2\right)}_{\text{Expected Squared Error}} = \underbrace{\left(\mathbf{E}\left(\hat{\theta}\right)-\theta\right)^2}_{\text{Bias Squared}} + \underbrace{\mathbf{E}\left(\left(\hat{\theta}-\mathbf{E}\left(\hat{\theta}\right)\right)^2\right)}_{\text{Variance}} \qquad (4)$$

If we want our estimator to be close to the truth on average (low expected squared error), then it is often worthwhile to accept some bias in order to reduce the variance of the estimator.

The bias-variance tradeoff is especially relevant in settings where the variance of an unbiased estimator is high. High-variance estimators are common when the number of parameters to be estimated (e.g., the means of many classes) is large, because the amount of data relevant to each parameter (e.g., the students in a particular class) may be small even if the overall sample size is very large. Beyond the setting of students in classrooms, the bias-variance tradeoff plays a central role in other statistical problems characterized by large sample sizes but also many parameters to be estimated, such as in small-area estimation (Rao, 2003). In machine learning, the bias-variance tradeoff is especially important because machine learning estimators often involve many parameters, such that variance is a serious concern even in big-data settings. Machine learning estimators resolve this problem by accepting some bias in order to reduce variance and improve expected squared error. Social scientists applying these methods should be comfortable with this acceptance of bias just as they are already comfortable with bias in classical statistical settings like multilevel models. The existence of bias should not be a barrier to the adoption of machine learning.

3.3 Data-driven estimator selection: Automate what can be automated

Choices abound in quantitative social science. For example, the choice of a model specification is a central question in classical statistics. Researchers have traditionally approached this question by some combination of conceptual argument paired with empirical metrics of model fit, such as R^2. A machine learning perspective transfers the weight of these choices in the direction of empirical evidence. To the degree that data can inform the choice of estimator, machine learning approaches allow the data to speak.

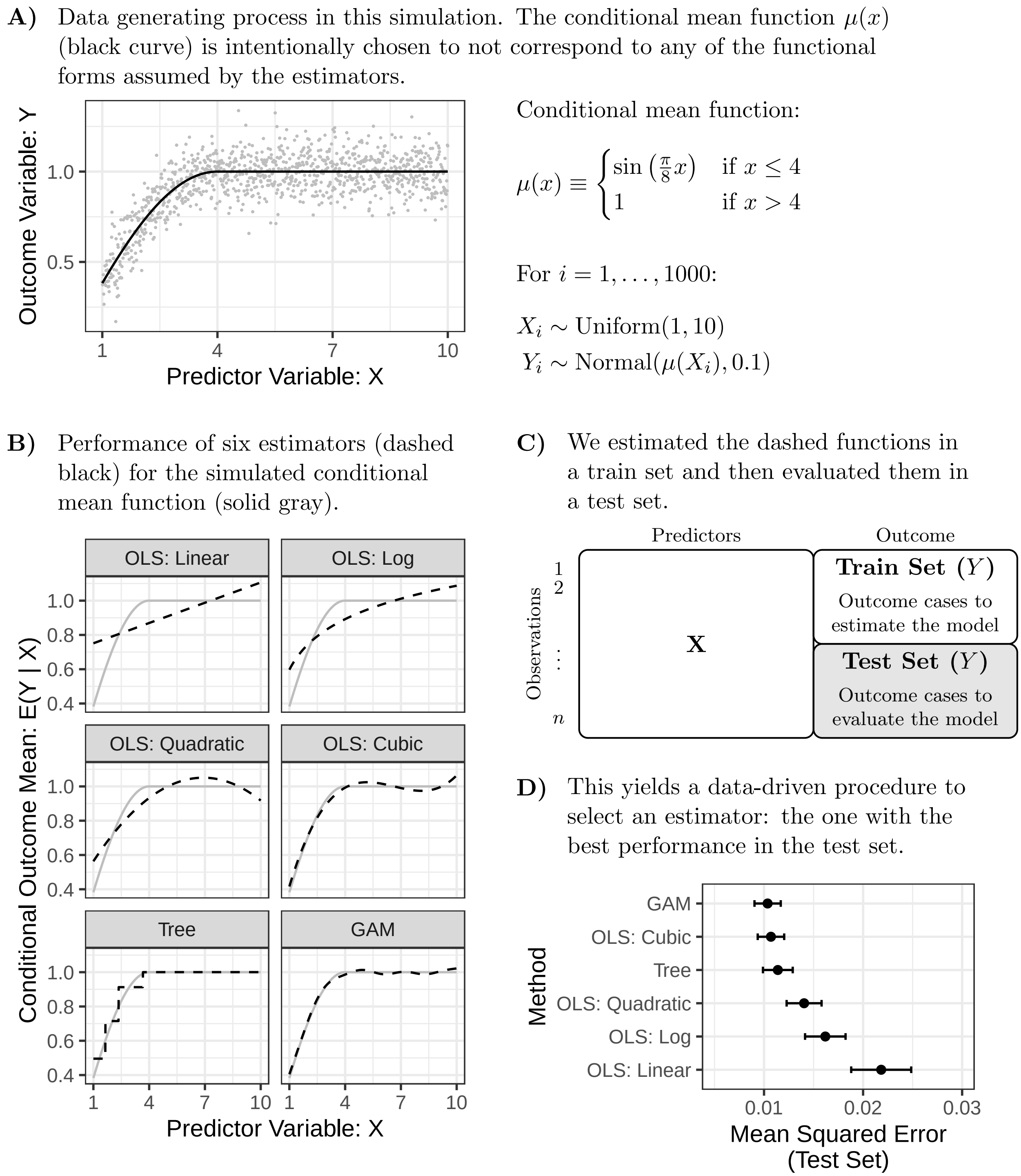
Fig 4 illustrates data-driven estimator selection in a simulated setting. The predictor variable X is related to the outcome Y by a complicated conditional mean function, as is likely to be the case in many realistic settings. Not knowing this function in advance, the researcher might consider several possible estimators with different assumed functional forms (various OLS specifications) or different procedures to learn the functional form from the data (a regression tree and a Generalized Additive Model). A social scientist following common practice might report the results of all these specifications. Despite the inclusion of machine learning estimators like regression trees, this overall research approach could be considered "classical" in the sense that it involves choosing the estimator or estimators for conceptual rather than data-driven reasons. An approach more inspired by machine learning might instead seek to empirically score the performance of the estimators in order to make a data-driven choice. The metric by which an estimator is evaluated is often called a *loss function*, which formalizes what it means for an estimator to perform well. For instance, one loss function would take an estimator $\hat{\theta}_\mathcal{S}$ estimated in a sample $\mathcal{S}$ and score it by its mean squared error when predicting new observations from the population.

$$\text{Loss Function:} \qquad \mathcal{L}(\hat{\theta}_\mathcal{S}) = \mathbf{E}_{i:i\notin\mathcal{S}}\left(\left(\hat{\theta}_\mathcal{S}(x_i) - y_i\right)^2\right) \tag{5}$$

In practice, we do not observe the full population and thus must rely on an estimate $\hat{\mathcal{L}}()$ of the loss function. Suppose we take our sample $\mathcal{S}$ and randomly assign observations into two equally-sized samples: a training sample $\mathcal{S}_\text{Training}$ and a test sample $\mathcal{S}_\text{Test}$ (Fig 4 Panel C). We then learn the prediction function in the training sample and estimate the loss function in the test sample.

$$\text{Estimated Loss Function:} \qquad \hat{\mathcal{L}}(\hat{\theta}_\mathcal{S}) = \frac{1}{|\mathcal{S}_\text{Test}|}\sum_{i\in\mathcal{S}_\text{Test}}\left(\hat{\theta}_{\mathcal{S}_\text{Training}}(x_i) - y_i\right)^2 \tag{6}$$

Finally, we can choose the estimator for which the estimated loss function $\hat{\mathcal{L}}(\hat{\theta}_\mathcal{S})$ is as close as possible to zero. In the simulated example of Fig 4, this procedure selects the Generalized Additive Model estimator. In this setting, it is visually apparent in Fig 4 Panel B that this is the best estimator. But in non-simulated settings, the true conditional mean function (the gray curve in



A) Data generating process in this simulation. The conditional mean function $\mu(x)$ (black curve) is intentionally chosen to not correspond to any of the functional forms assumed by the estimators.

Conditional mean function:

$$\mu(x) \equiv \begin{cases} \sin\left(\frac{\pi}{8}x\right) & \text{if } x \leq 4 \\ 1 & \text{if } x > 4 \end{cases}$$

For $i = 1, \ldots, 1000$:

$$X_i \sim \text{Uniform}(1, 10)$$
$$Y_i \sim \text{Normal}(\mu(X_i), 0.1)$$

B) Performance of six estimators (dashed black) for the simulated conditional mean function (solid gray).

C) We estimated the dashed functions in a train set and then evaluated them in a test set.

D) This yields a data-driven procedure to select an estimator: the one with the best performance in the test set.

Fig. 4. Simulation: Data-driven estimator selection. We consider six estimators: OLS with linear, log, quadratic, and cubic specifications, a regression tree following defaults in the `rpart` package (Therneau et al., 2015), and a Generalized Additive Model (GAM, Wood 2017) following defaults in the `mgcv` package. Visually, the GAM comes closest to the true response function (Panel B). Panel C depicts how we randomly assigned observations to two equally-sized subsamples: the train set and test set. We then estimated each function on the train set and estimated its mean squared error when predicting the new cases in the test set. Panel D shows that the GAM achieves the best performance. This exercise illustrates a building block of machine learning: instead of arguing conceptually for a particular estimator (e.g. OLS with a particular form), empirically evaluate the performance of many candidate estimators.

Fig 4 Panel B) is unknown and out-of-sample predictive performance can still help the researcher choose an estimator which comes as close as possible to that unknown function.

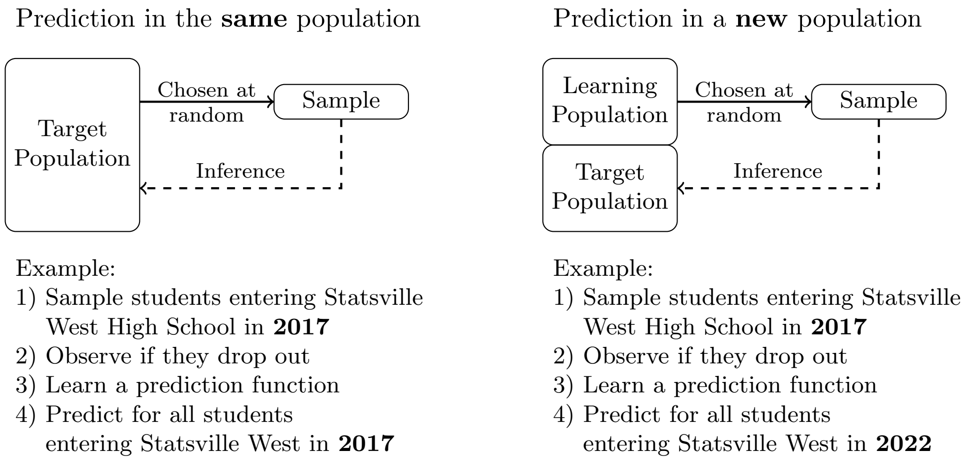
While data-driven estimator selection is a hallmark of machine learning, it is also in full alignment with standard statistical procedures. Social scientists already compare models by empirical scores such as R^2, likelihood ratios, the Akaike information criterion (AIC, Akaike 1973), Bayesian information criterion (BIC, Schwarz 1978), and numerous other scores. Each of these can be interpreted as a loss function for data-driven model selection. When carried out within machine learning, the loss function is typically evaluated on data not used to estimate the model in order to assess the ability of the model to generalize to new observations.

We have taken care to distinguish the true loss function $\mathcal{L}(k)$ from the estimated loss function $\hat{\mathcal{L}}(k)$ because the estimated loss function may be statistically uncertain, especially if it evaluated on a small sample. An estimator which is inferior in the population may outperform another estimator in the test sample because of the chance of which cases from the population happen to appear in the test sample. One way to improve the precision of $\hat{\mathcal{L}}(k)$ is to conduct cross validation, a procedure in which the full sample $\mathcal{S}$ is partitioned into several components, each of which plays the role of $\mathcal{S}_{\text{Test}}$ in turn, with the ultimate loss function estimate being the average of the results. Regardless of whether a single sample split or cross validation is used to select a model, researchers should be cautious of the possibility that the estimated loss function may itself be statistically uncertain, so that the evidence for one model over another may be weaker than point estimates alone might suggest.

3.4 A return to task clarity: Use caution when predicting in a new target population

Our first conceptual building block was task clarity—being precise about goal of the quantitative exercise. To re-emphasize the importance of task clarity, we now turn from standard out-of-sample prediction tasks to a range of more complex tasks. We discuss two settings that demonstrate the importance of clarity about the task: prediction in a new target population and prediction for causal inference.

To consider prediction in a new target population, suppose we study a cohort of students entering Statsville West High School in 2017. For each student, we observe many variables about



Prediction in the **same** population

Example:
1) Sample students entering Statsville West High School in **2017**
2) Observe if they drop out
3) Learn a prediction function
4) Predict for all students entering Statsville West in **2017**

Prediction in a **new** population

Example:
1) Sample students entering Statsville West High School in **2017**
2) Observe if they drop out
3) Learn a prediction function
4) Predict for all students entering Statsville West in **2022**

Fig. 5. Caution: Prediction in a new target population. A standard machine learning task is to learn about a target population using a sample of cases selected at random from that population. In practice, however, algorithms are often deployed to make predictions in new populations from which no training cases were available. For example, a function to predict high school dropout learned in a cohort entering high school in 2017 might be used to target resources to at-risk students entering high school in 2022. But if the mapping between the predictors and outcome changes across cohorts, that prediction function may no longer be useful. To the extent that prediction functions are learned in one population and applied in a new target population, the validity of predictions may be placed in doubt.

academic performance in 8th grade and we observe whether they drop out of high school over the next four years. Using a machine learning algorithm, we learn a function to predict high school dropout. Impressed by our model, the principal of Statsville West suggests that for the entering cohort of 2022 we predict the likelihood of dropping out for each student, so that the principal can target extra counseling resources to those students. Perhaps the principal of Statsville East High School also hears about our model and wants to deploy it in that context as well. For each of these use cases, there is a danger: the population that entered Statsville West in 2017 is not the same as the population entering in 2022, and is surely different from the population entering Statsville East in 2022. The mapping between the predictors and the outcome in these new populations may not be the same as the mapping in the original population on which the algorithm was learned (Statsville West, entering in 2017). The problem of Statsville West and Statsville East is ubiquitous across real applications of machine learning. Researchers routinely learn things in one context in the past, and then apply what they have learned in the future and possibly in new contexts. To use statistics and machine learning responsibly, one must be aware when there is a leap to a new target population (Fig 5).

Prediction in a new target population is especially relevant for causal inference (Fig 6). Suppose the principal of Statsville West had already implemented a program to offer extra counseling to some students in the 2017 entering cohort. After observing whether those students dropped out, the principal wants to predict whether those who did not receive the program would have benefited if the program had been available to them. But those who did not receive the extra counseling are by definition not part of the learning population from whom we drew the sample. In fact, it is impossible to sample people who received the counseling and observe the outcome they would have realized if they had not received the counseling (the fundamental problem of causal inference, Holland 1986). To learn about what would have happened if other students had received extra counseling, the principal is necessarily requiring the researcher to make predictions in a new population. Absent randomization or additional assumptions, prediction for causal questions *always* involves a target population which is different from the learning population. Only by an assumption can we view the learning population and the target population as the same population in causal inference. For instance, we might assume (or know from randomization) that the potential outcome under treatment $Y_i(1)$ follows the same distribution among the treated units as among the untreated units, within each subpopulation defined by a set of predictor values. By this assumption, any mapping $\vec{X}_i \rightarrow Y_i(1)$ learned in the learning population will still be valid in the target population.

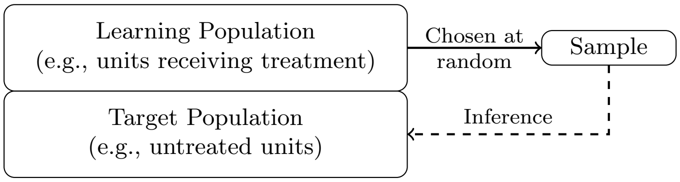
Yet even in the best-case scenario, causal inference for policy prescriptions often involves an additional leap to a new target population (Fig 7). Suppose the principal randomly assigned counseling to students entering Statsville West in 2017. But then, the principal wants to use these results to justify the an expansion of counseling support for the cohort entering in 2022. Despite strong internal validity for the causal effect estimate in the 2017 cohort, the principal still must leap to a new population to deploy the policy in the 2022 cohort. The leap from the training population to the target population is therefore particularly relevant to causal policy prescriptions.[4]

In fact, there is often a tradeoff between internal and external validity, where one can study a population less like the target population in a randomized design (high internal validity) or a population more like the target population in an observational study (high external validity).

[4] The assumption to draw causal inference in the target population is $\{Y(0), Y(1)\} \perp\!\!\!\perp \{\mathcal{P}, S, D\} \mid \vec{X}$, where $\mathcal{P}$ indicates membership in the learning versus target population, S indicates inclusion in the sample of cases, D indicates treatment assignment, and $\vec{X}$ denotes the vector of pre-treatment predictors. One setting where this would hold is if the target population is the learning population and S and D are randomly assigned.

Prediction in a **counterfactual** population.

Task: Predict the outcome that would be realized under treatment.



Required Assumption: Treatment (which determines membership in the learning versus target population) is independent of the outcome $Y_i(1)$ that would be realized under treatment, within subpopulations defined by the predictor variables.

Fig. 6. Causal inference: A task that involves a new target population. Suppose we observe a set of units who receive a treatment of interest (e.g., extra counseling in high school). After learning a prediction function in a sample of treated units, we wish to predict the outcome that untreated units would have realized if they had received treatment. For instance, we might predict whether those who did not receive counseling would not have dropped out if they had received counseling. Causal questions of this form require assumptions because in the absence of a randomized treatment it is impossible to draw a simple random sample from the target population.

Perhaps the principal has a randomized experiment from a very old cohort that entered in 2000 and an observational study on the cohort that entered in 2017. It would not be clear which study would be more informative for a policy prescription applying to the cohort entering in 2022. Every study has limitations, and the leap from a learning population to a different target population is a limitation of which one must always be aware.

4 The future: Guiding principles and promising approaches

Looking ahead, it is difficult to predict how machine learning will be used in the future. In this section, we offer a few guiding principles and approaches which we believe may hold particular promise for future use in social science.

4.1 Resolving p-hacking: The promise of automated model selection

The replication crisis has cast doubt on the validity of much quantitative social science research (Freese and Peterson, 2017; Simmons et al., 2011). A key source of concern is the common practice in which researchers iterate between model fitting and interpretation until arriving at a chosen specification which is reported to the reader. This procedure creates many opportunities for a well-

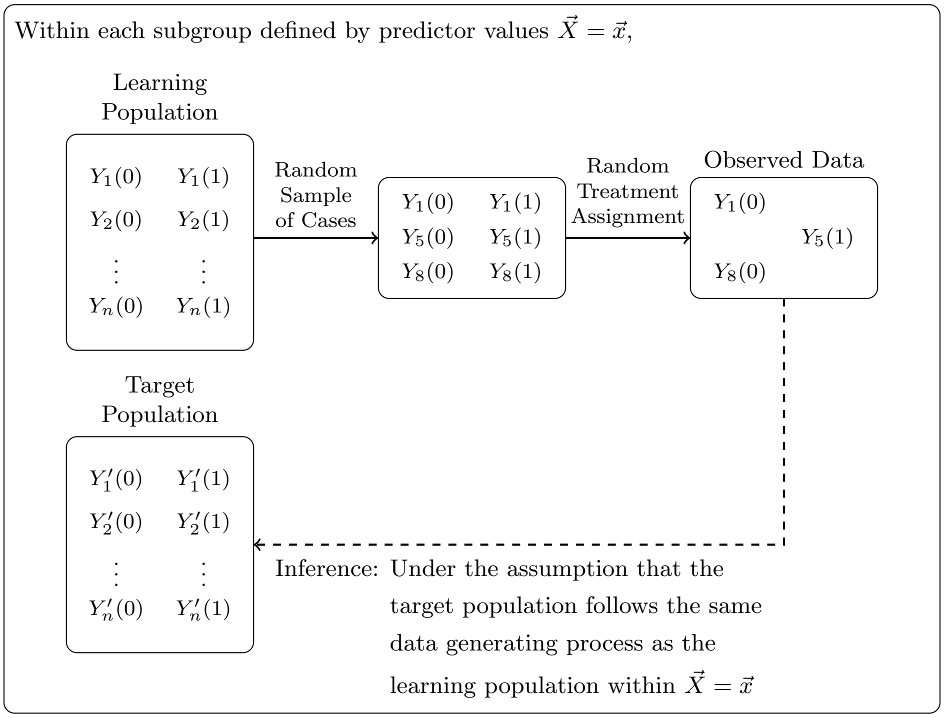


Fig. 7. Causal inference for policy prescriptions: A particular leap to a new target. Suppose there is a learning population of n units, each of whom has a potential outcome that would be realized under the control condition $Y_i(0)$ and under the treatment condition $Y_i(1)$. Suppose we take a random sample from the learning population and then randomly assign treatments to units in that sample. For each unit in the sample, we observe one of the two potential outcomes. Under randomization, a prediction function learned in the observed data can be used to make predictions in the learning population. But when designing policy, we generally want to predict treatment effectiveness in a new population who have not yet received the treatment. To predict in a new target population, we would have to additionally assume that the same data generating process holds in the target population as in the learning population. In observational settings, machine learning can be used for causal inference if the assumptions of random sampling and random treatment assignment are credible within subgroups defined by the observed predictors $\vec{X}$.

meaning researcher to select the model for which the results that most aligns with that researcher's pre-existing beliefs (Gelman and Loken, 2013). The (possibly unintentional) practice of choosing an estimator based on the results undermines the validity of p-values and confidence intervals, which are designed under the assumption that the researcher follows a single procedure that would be applied the same way in any hypothetical sample.

Machine learning may seem to amplify this problem: with more candidate estimators, researchers who stay the course will simply have more opportunities to select their preferred result. Automated model selection offers a way out of this problem. Before analyzing the data, one can specify a single decision rule for choosing among many candidate estimators. For instance, we might choose the one with the lowest cross-validated mean squared error. By defining the decision rule before viewing any results, one can remove the danger of choosing a result based on one's preferred specification. One particularly promising application of automated model selection is Super Learner (Van der Laan et al., 2007), which accepts a dataset and a set of candidate learners as arguments and returns a single prediction function which is a weighted average of those learners with weights learned through cross-validation. Super Learner is available in open-source software for R, both in the `SuperLearner` package (Van der Laan et al., 2007) and as part of the `tlverse` in the `sl3` package (Coyle et al., 2021).

4.2 Resolving approximate models: The promise of an agnostic perspective

It is often stated that all models are wrong. Yet when a researcher assumes a statistical model, the properties of that model depend on the assumptions it entails. As Manski (2003) argued, the credibility of the resulting inference is only as strong as the researcher's defense of the required assumptions. Yet social scientists routinely assume linear, additive models despite conceptual reasons to expect the world to be nonlinear and interactive. When a model is a poor approximation to the world, a hypothesis about some coefficient β may be of limited use.

As with replication, machine learning may seem to make this problem worse. It is one thing to argue about whether an Ordinary Least Squares model is correctly specified; it is quite another to argue that a random forest or deep neural network is correctly specified. A promising domain of statistical research reinterprets linear regression parameters as estimators of the best linear approximation to the conditional mean function (Aronow and Miller, 2019; Buja et al., 2019a,b; Lin, 2013), but producing a similar interpretation for complex machine learning estimators is likely out of reach. Because machine learning estimators may be harder to interpret than classical statistical estimators, the difficulty of assuming one's model to be correct may become even more severe.

An alternative agnostic approach resolves this problem by doing away with the notion of "correct" model specification altogether. The world is complex, and every statistical or machine learning model is likely to be an approximation at best. A researcher who defines the estimand outside of the statistical model gains an opportunity to be transparent about the sense in which the model is an approximation (Lundberg et al., 2021). Then, predictive performance can be used to assess the relative accuracy of various candidate approximations. New tools for estimation may produce an approximation which is closer to the truth than the standard model-based approximations. We therefore join others in arguing that social scientists adopting machine learning tools would do well to maintain an agnostic perspective, where the research goal is defined outside of the model and the parameters of the model are only tools to approximate that target quantity (Grimmer et al., 2021).

4.3 Resolving extrapolation: The promise of local estimators

Extrapolation is an ever-present danger in globally parametric models like Ordinary Least Squares. Two complementary sides of this same problem are extrapolation and influence. Extrapolation occurs when a data point to be predicted is far from the mass of the training data, so that the predicted value may depend heavily on the assumed functional form (e.g., a line). Influence is the converse, when a training point far from the mass of the data heavily shapes the fitted prediction function. Extrapolation and influence are two consequences with the same source: the assumption of global parametric models (e.g., the assumption of a line). Local estimators offer a solution to the problem: only allow each unit j to contribute to the estimate for unit i to the degree that unit j is "near" to unit i. For every local estimator, the central question is what it means for two units to be "near" each other. New advances in local estimation are thus most powerful paired with conceptual social science argument for the chosen definition of "near."

Propensity score matching for causal inference is one example of a local estimator (Imbens, 2015; Morgan and Harding, 2006). Suppose we know the probability of treatment p_i (also known as the propensity score) given the values of confounding variables for unit i. If unit i is treated, we might estimate the potential outcome under control $Y_i(0)$ by the outcome of the untreated unit j with propensity score p_j closest to unit i. This is a local estimator because only the nearest untreated unit contributes to the estimate for unit i. Propensity score matching is a nearest neighbors estimator (Fix and Hodges, 1951): the unit or units nearest to the focal unit contribute to the estimate.

Nearest neighbors and other local estimators depend crucially on the definition of "near." There are many ways to define what it means to be near. In propensity score matching, the distance between any pair of units is defined as the difference in their probabilities of treatment p_i, each of which is a univariate summary of the confounder set $\vec{L}_i$. But one could also define nearness as a function of the confounders $\vec{L}_i$ directly, as is the case for Manhattan distance (sum of absolute differences over all in covariate values), Euclidean distance (sum of squared differences), or Mahalanobis distance (a generalization of Euclidean distance which incorporates the covariance among $\vec{X}$, Mahalanobis 1936). For each of these distances, one can define a local estimator by averaging across units which are "near" the focal unit by the chosen distance metric.

The definition of nearness is consequential: units that are "near" by one metric may be far apart by another metric.[5] Future research with local estimators will need to reason carefully about the definition of "near" that is relevant to the problem at hand. For instance, the covariate balancing propensity score (CBPS) (Imai and Ratkovic, 2014) modifies the propensity score to optimize balance along the covariates. Entropy balancing (Hainmueller, 2012) optimizes matches such that first, second, or higher moments of the covariates are similar across matched units. Coarsened exact matching (Iacus et al., 2012) defines units to be "near" if and only if they take identical values along a coarsened version of measured covariates, but this distance metric is ambivalent about differences in covariate values within the coarsened strata. No distance metric is inherently superior to another outside of a specific application—they are all different definitions of what it means to be "near."

Machine learning tools offer new ways to define the distance between any pair of observations. Random forests (Breiman, 2001) are one example. A random forest is an algorithm which repeatedly (1) randomly samples a subset of predictors from the data, (2) randomly samples observations from

[5] In fact, when the predictor set $\vec{L}$ is high-dimensional (containing many unique values), it is possible that every unit is in some sense quite far from all other units. In causal inference, this can create a setting where arguably there is no untreated unit which is comparable to any given treated unit (D'Amour et al., 2021).

the data with replacement, and (3) partitions the resulting sample into a set of "leaves" which are cells defined by the predictor variables and for which the outcome Y is relatively homogeneous. Each iteration produces a tree, and the average of all the trees is a forest. As highlighted by Lin and Jeon (2006), the random forest can be interpreted as a weighted nearest neighbors estimator, where units i and j are "near" to each other proportional to the frequency that they fall in the same leaf. The connection is powerful because it connects random forests (a machine learning tool) to a setting well-studied in classical statistics (weighted means). Wager and Athey (2018) exploit this connection to derive asymptotically-valid confidence intervals for estimates from random forests, drawing on results from classical statistics (Hájek, 1968; Hoeffding, 1948). The notion of random forests as adaptive nearest neighbors estimator generalizes to many problems (Athey et al., 2019), such as using random forests to define nearness for weighted local linear regression (Friedberg et al., 2021).

Looking forward, local estimators hold great promise for future social science research. The barriers to adoption are low: many of the advances discussed above are implemented in open-source R software, including `cbps` for the covariate balancing propensity score (Fong et al., 2021), `ebal` for entropy balancing (Hainmueller, 2014), `ranger` for random forests (Wright and Ziegler, 2017), and `grf` for generalized random forests (Tibshirani et al., 2018). The open task for social scientists is to motivate the chosen definition of "near" with respect to their substantive problem.

4.4 Resolving poor convergence: The promise of targeted learning

Flexible machine learning estimators such as random forests can approximate unknown conditional mean functions $\mathbf{E}(Y \mid \vec{X})$ without the strong parametric assumptions common in to classical methods like generalized linear models. Yet flexibility comes at a cost: the rate at which adaptive estimators converge toward the conditional mean is slower than the rate achieved by parametric methods. Targeted learning (Van der Laan and Rose, 2018; Van Der Laan and Rubin, 2006) resolves this convergence problem. While one cannot generally achieve fast convergence for the full conditional mean function, it is often possible to target the estimator and achieve fast convergence rates for a low-dimensional parameter of social science interest.

For concreteness, suppose we are interested in the population-average potential outcome $\mathbf{E}(Y(a))$ that would be realized if a treatment variable A were assigned to the value a. This target parameter is *low-dimensional*: in this case, it is just one number. We might make the causal assumption that a set of measured confounders $\vec{L}$ is sufficient to identify that causal parameter. In this case, our causal target parameter can be rewritten as a particular aggregation of a statistical function.

$$\overset{\substack{\text{Expected outcome}\\ \text{under treatment } a\\ \textbf{(low-dimensional)}\\ \downarrow}}{\mathbf{E}(Y(a))} = \underset{\substack{\uparrow\\ \text{Outer expectation over the} \quad \text{Converts a high-dimensional function}\\ \text{population distribution of } \vec{L} \quad \text{to a low-dimensional target parameter}}}{\mathbf{E}}\left(\overset{\substack{\text{Conditional mean}\\ \textbf{(high-dimensional)}\\ \downarrow}}{\mathbf{E}}\left(Y \mid A = a, \vec{L}\right)\right) \tag{7}$$

The internal conditional expectation $\mathbf{E}(Y \mid A = a, \vec{L})$ is a *high-dimensional parameter* because the confounders $\vec{L}$ may have many unique values. The reason flexible machine learning estimators have slow convergence is because they attempt to estimate all of these conditional means under minimal assumptions. But in our setting, the only reason for estimating the high-dimensional parameter is to help us estimate the low-dimensional target that we really want.

Targeted learning takes advantage of the aggregation to target the estimator to our specific goal. In this setting (see Fig 8), we would begin by estimating a prediction function $\hat{g}(a, \vec{L}) \approx \mathbf{E}(Y \mid a, \vec{L})$ to approximate the internal conditional expectation function. For instance, we might restrict to the subpopulation with treatment value $A = a$ and then predict the outcome Y as a function of the confounders $\vec{L}$. Then, we might use that prediction function to make predictions in the full population. Yet here is a problem: suppose a stratum $\vec{L} = \vec{\ell}$ of the confounders contains only a few treated units ($A = a$) but also contains many untreated units ($A \neq a$). A naive prediction function will not optimize for prediction in this stratum, because few treated units are observed in the stratum. Yet, in the target population this stratum may be very important because it is home to many untreated units. We ideally desire an estimator which will perform well in the places we want to make predictions, which may not be the places where we already have plentiful data. Naive prediction optimizes for the wrong task.

To target our estimator, we need information about how the confounding variables $\vec{L}$ are related to the treatment condition $A = a$. We could estimate the conditional probability of treatment $\hat{m}(a, \vec{L}) \approx \mathrm{P}(A = a \mid \vec{L})$. The inverse of this conditional probability for each treated unit is

1) Estimate initial prediction functions

$$\hat{g}(a,\vec{\ell}) \approx \mathrm{P}\left(Y = 1 \mid A = a, \vec{L} = \vec{\ell}\right)$$
$$\hat{m}(a,\vec{\ell}) \approx \mathrm{P}\left(A = a \mid \vec{L}\right)$$

2) Define the clever covariate

$$\hat{H}(A,\vec{L}) = \tfrac{\mathbb{I}(A=a)}{\hat{m}(a,\vec{L})}$$

Sample split: Optionally, carry out step 3 in a different sample from steps 1 and 2

3) Regress Y on the clever covariate with an offset

$$\operatorname{logit}\left(\mathrm{P}(Y = 1 \mid A, \vec{L})\right) \approx \underset{\substack{\uparrow \\ \text{Offset} \\ \text{term} \\ \text{(from 1)}}}{\operatorname{logit}}\left(\hat{g}(A,\vec{L})\right) + \underset{\substack{\nearrow \\ \text{Clever} \\ \text{covariate} \\ \text{(from 2)}}}{\hat{H}}(A,\vec{L})\underset{\substack{\uparrow \\ \text{Coefficient} \\ \text{to estimate} \\ \text{here}}}{\beta}$$

4) Target the prediction function

$$\underset{\substack{\uparrow \\ \text{Targeted prediction rule} \\ \text{optimized for the way we} \\ \text{will aggregate predictions}}}{\hat{g}'(A,\vec{L})} = \operatorname{logit}^{-1}\left(\underset{\substack{\uparrow \\ \text{Original prediction rule} \\ \text{optimized for} \\ \text{disaggregate prediction}}}{\operatorname{logit}(\hat{g}(A,\vec{L}))} + \hat{H}(A,\vec{L})\hat{\beta}\right)$$

5) Estimate using the targeted prediction function

$$\hat{\mathbf{E}}(Y(a)) = \tfrac{1}{n}\textstyle\sum_{i=1}^{n} \hat{g}'(a,\vec{L}_i)$$

Fig. 8. Targeted learning with a binary outcome. An important advantage of targeted learning (Van der Laan and Rose, 2018) over double machine learning (Chernozhukov et al., 2018) is that targeted learning can accomodate a link function (the logit in steps 3 and 4) which can guarantee that predicted values fall within the support of the outcome.

proportional to the ratio of units with this confounder set in the full population (treated and untreated) to the number with this confounder set in the treated population alone. It tell us how much a unit like this is overrepresented in our target population compared with our available sample. If there is a trend such that our outcome model over- or under-predicts in strata that will be heavily upweighted when moving from the treated population to the full population, we want to correct that outcome model. To do that, one can estimate a new regression for Y where the initial prediction $\hat{g}(A, \vec{L})$ is included as an offset term (a known intercept) and the inverse treatment probability $\frac{\mathbb{I}(A_i=a)}{\hat{m}(a,\vec{L})}$ is included as the "clever covariate." The estimated coefficient $\hat{\beta}$ on the clever covariate captures the degree to which outcomes tend to be over- or under-predicted in strata that will be heavily weighted for inference in the population. Using that estimated coefficient, we arrive at a new prediction function $\hat{g}'(a, \vec{L}_i)$ which we then use to predict the potential outcome under treatment $A = a$ for every unit i in the target population. The average of those predictions is a targeted estimate of the population average potential outcome, $\mathbf{E}(Y(a))$.

Targeted learning should be more widely applied in social science because it offers several important advantages. First, the convergence rate for the target parameter (in this case, $\mathbf{E}(Y(a))$) is faster than the convergence rates for the high-dimensional prediction functions $\hat{g}(a, \vec{\ell})$ and $\hat{m}(a, \vec{\ell})$. This property allows one to use flexible machine learning estimators such as random forests (Breiman, 2001) and ensemble methods such as super learner (Van Der Laan and Dudoit, 2003), even though these estimators have slow convergence rates. Second, because targeting (Step 4 in Fig 8) can involve a generalized linear model with a link function designed to match the support of the outcome variable, targeted learning never makes predictions outside of that support. This property is not shared by other related methods, such as augmented inverse probability weighting (Robins and Rotnitzky, 1995; Robins et al., 1994) and double machine learning (Chernozhukov et al., 2018).[6] Third, targeted learning is a general-purpose methodology which can be applied to many settings, while maintaining formal properties such as consistency and asymptotic normality which derive from the roots of the method in influence functions (Van der Laan and Rose, 2018). Finally, targeted learning is accessible: for common target parameters, the `tlverse` suite of R packages supports the use of targeted learning.

[6] Appendix Fig 10 presents Double Machine Learning (Chernozhukov et al., 2018), and Appendix Fig 9 presents targeted learning ofr a continuous outcome to support direct comparisons with DMl.

5 Conclusion

To be written :)

References

Akaike, H. (1973). *Information theory and the maximum likelihood principle.* Budapest: Akademiai Kiado.

Aronow, P. M. and Miller, B. T. (2019). *Foundations of Agnostic Statistics.* Cambridge University Press.

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

Athey, S., Tibshirani, J., and Wager, S. (2019). Generalized random forests. *The Annals of Statistics*, 47(2):1148–1178.

Bisbee, J. (2019). Barp: Improving mister p using bayesian additive regression trees. *American Political Science Review*, 113(4):1060–1065.

Brand, J. E., Xu, J., Koch, B., and Geraldo, P. (2021). Uncovering sociological effect heterogeneity using tree-based machine learning. *Sociological Methodology*, 51(2):189–223.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3):199–231.

Breitenstein, S. (2019). Choosing the crook: A conjoint experiment on voting for corrupt politicians. *Research & Politics*, 6(1):2053168019832230.

Bryk, A. S. and Raudenbush, S. W. (1992). *Hierarchical Linear Models: Applications and Data Analysis Methods.* Sage Publications, Inc.

Buja, A., Brown, L., Berk, R., George, E., Pitkin, E., Traskin, M., Zhang, K., and Zhao, L. (2019a). Models as approximations i: Consequences illustrated with linear regression. *Statistical Science*, 34(4):523–544.

Buja, A., Brown, L., Kuchibhotla, A. K., Berk, R., George, E., and Zhao, L. (2019b). Models as approximations ii: A model-free theory of parametric regression. *Statistical Science*, 34(4):545–565.

Cantú, F. (2019). The fingerprints of fraud: Evidence from Mexico's 1988 presidential election. *American Political Science Review*, 113(3):710–726.

Chalfin, A., Danieli, O., Hillis, A., Jelveh, Z., Luca, M., Ludwig, J., and Mullainathan, S. (2016). Productivity and selection of human capital with machine learning. *American Economic Review*, 106(5):124–27.

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

Coyle, J., Hejazi, N., Malenica, I., Phillips, R., and Sofrygin, O. (2021). *sl3: Pipelines for Machine Learning and Super Learning.* R package version 1.4.4.

Davis, J. and Heller, S. B. (2017). Using causal forests to predict treatment heterogeneity: An application to summer jobs. *American Economic Review*, 107(5):546–50.

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

D'Amour, A., Ding, P., Feller, A., Lei, L., and Sekhon, J. (2021). Overlap in observational studies with high-dimensional covariates. *Journal of Econometrics*, 221(2):644–654.

Fix, E. and Hodges, J. L. (1989[1951]). Discriminatory analysis. nonparametric discrimination: Consistency properties. *International Statistical Review/Revue Internationale de Statistique*, 57(3):238–247.

Fong, C., Ratkovic, M., Imai, K., Hazlett, C., Yang, X., Peng, S., and Lee, I. (2021). *CBPS: Covariate Balancing Propensity Score.* R package version 0.23.

Freese, J. and Peterson, D. (2017). Replication in social science. *Annual Review of Sociology*, 43:147–165.

Friedberg, R., Tibshirani, J., Athey, S., and Wager, S. (2021). Local linear forests. *Journal of Computational and Graphical Statistics*, 30(2):503–517.

Friedman, S. and Reeves, A. (2020). From aristocratic to ordinary: Shifting modes of elite distinction. *American Sociological Review*, 85(2):323–350.

Gelman, A. and Little, T. C. (1997). Poststratification into many categories using hierarchical logistic regression. *Survey Methodology*, 23(2):127–135.

Gelman, A. and Loken, E. (2013). The garden of forking paths: Why multiple comparisons can be a problem, even when there is no "fishing expedition" or "p-hacking" and the research hypothesis was posited ahead of time.

Grimmer, J., Roberts, M. E., and Stewart, B. M. (2021). Machine learning for social science: An agnostic approach. *Annual Review of Political Science*, 24:395–419.

Hainmueller, J. (2012). Entropy balancing for causal effects: A multivariate reweighting method to produce balanced samples in observational studies. *Political Analysis*, 20(1):25–46.

Hainmueller, J. (2014). *ebal: Entropy reweighting to create balanced samples.* R package version 0.1-6.

Hainmueller, J., Hopkins, D. J., and Yamamoto, T. (2014). Causal inference in conjoint analysis: Understanding multidimensional choices via stated preference experiments. *Political Analysis*, 22(1):1–30.

Hájek, J. (1968). Asymptotic normality of simple linear rank statistics under alternatives. *The Annals of Mathematical Statistics*, pages 325–346.

Handel, B. and Kolstad, J. (2017). Wearable technologies and health behaviors: new data and new methods to understand population health. *American Economic Review*, 107(5):481–85.

Hoeffding, W. (1948). A class of statistics with asymptotically normal distribution. *The Annals of Mathematical Statistics*, 19(3):293–325.

Holland, P. W. (1986). Statistics and causal inference. *Journal of the American Statistical Association*, 81(396):945–960.

Hopkins, D. J. and King, G. (2010). A method of automated nonparametric content analysis for social science. *American Journal of Political Science*, 54(1):229–247.

Iacus, S. M., King, G., and Porro, G. (2012). Causal inference without balance checking: Coarsened exact matching. *Political Analysis*, 20(1):1–24.

Imai, K. and Ratkovic, M. (2014). Covariate balancing propensity score. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(1):243–263.

Imbens, G. W. (2015). Matching methods in practice: Three examples. *Journal of Human Resources*, 50(2):373–419.

Imbens, G. W. and Rubin, D. B. (2015). *Causal Inference in Statistics, Social, and Biomedical Sciences*. Cambridge University Press.

Incerti, T. (2020). Corruption information and vote share: A meta-analysis and lessons for experimental design. *American Political Science Review*, 114(3):761–774.

Jerzak, C. T., King, G., and Strezhnev, A. (2019). An improved method of automated nonparametric content analysis for social science. *Political Analysis*, pages 1–17.

King, G., Pan, J., and Roberts, M. E. (2017). How the Chinese government fabricates social media posts for strategic distraction, not engaged argument. *American Political Science Review*, 111(3):484–501.

Kleinberg, J., Ludwig, J., Mullainathan, S., and Obermeyer, Z. (2015). Prediction policy problems. *American Economic Review*, 105(5):491–95.

Knox, D. and Lucas, C. (2021). A dynamic model of speech for the social sciences. *American Political Science Review*, 115(2):649–666.

Lin, W. (2013). Agnostic notes on regression adjustments to experimental data: Reexamining freedman's critique. *The Annals of Applied Statistics*, 7(1):295–318.

Lin, Y. and Jeon, Y. (2006). Random forests and adaptive nearest neighbors. *Journal of the American Statistical Association*, 101(474):578–590.

Lundberg, I., Johnson, R., and Stewart, B. M. (2021). What is your estimand? Defining the target quantity connects statistical evidence to theory. *American Sociological Review*, 86(3):532–565.

Mahalanobis, P. C. (1936). On the generalized distance in statistics. National Institute of Science of India.

Manski, C. F. (2003). *Partial Identification of Probability Distributions*, volume 5. Springer.

Morgan, S. L. and Harding, D. J. (2006). Matching estimators of causal effects: Prospects and pitfalls in theory and practice. *Sociological Methods & Research*, 35(1):3–60.

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

Rao, J. N. (2003). *Small Area Estimation*. John Wiley & Sons.

Robins, J. M. and Rotnitzky, A. (1995). Semiparametric efficiency in multivariate regression models with missing data. *Journal of the American Statistical Association*, 90(429):122–129.

Robins, J. M., Rotnitzky, A., and Zhao, L. P. (1994). Estimation of regression coefficients when some regressors are not always observed. *Journal of the American statistical Association*, 89(427):846–866.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, pages 461–464.

Simmons, J. P., Nelson, L. D., and Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, 22(11):1359–1366.

Su, Z. and Meng, T. (2016). Selective responsiveness: Online public demands and government responsiveness in authoritarian china. *Social Science Research*, 59:52–67.

Szeliski, R. (2010). *Computer vision: algorithms and applications*. Springer Science & Business Media.

Therneau, T., Atkinson, B., Ripley, B., and Ripley, M. B. (2015). Package 'rpart'. *Available online: https://cran.r-project.org/web/packages/rpart/index.html*.

Tibshirani, J., Athey, S., Friedberg, R., Hadad, V., Hirshberg, D., Miner, L., Sverdrup, E., Wager, S., Wright, M., and Tibshirani, M. J. (2018). Package 'grf'.

Van Der Laan, M. J. and Dudoit, S. (2003). Unified cross-validation methodology for selection among estimators and a general cross-validated adaptive epsilon-net estimator: Finite sample oracle inequalities and examples.

Van der Laan, M. J., Polley, E. C., and Hubbard, A. E. (2007). Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6(1).

Van der Laan, M. J. and Rose, S. (2018). *Targeted Learning in Data Science: Causal Inference for Complex Longitudinal Studies*. Springer.

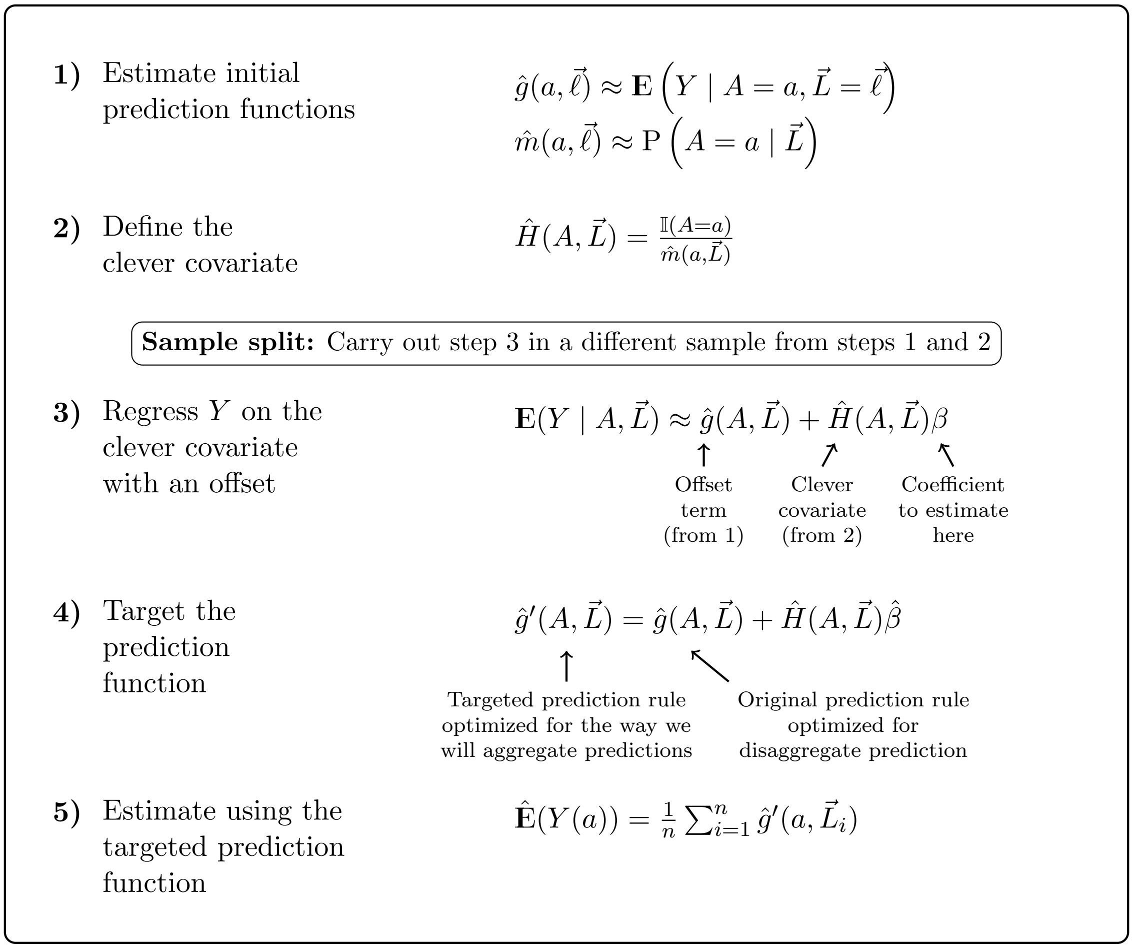
Van Der Laan, M. J. and Rubin, D. (2006). Targeted maximum likelihood learning. *The international Journal of Biostatistics*, 2(1).

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

Wood, S. N. (2017). *Generalized Additive Models: An Introduction with R*. CRC press.

Wright, M. N. and Ziegler, A. (2017). ranger: A fast implementation of random forests for high dimensional data in c++ and r. *Journal of Statistical Software*, 77(1):1–17.

A Supplemental figures



1) Estimate initial prediction functions

$$\hat{g}(a,\vec{\ell}) \approx \mathbf{E}\left(Y \mid A = a, \vec{L} = \vec{\ell}\right)$$

$$\hat{m}(a,\vec{\ell}) \approx \mathrm{P}\left(A = a \mid \vec{L}\right)$$

2) Define the clever covariate

$$\hat{H}(A,\vec{L}) = \frac{\mathbb{I}(A=a)}{\hat{m}(a,\vec{L})}$$

Sample split: Carry out step 3 in a different sample from steps 1 and 2

3) Regress Y on the clever covariate with an offset

$$\mathbf{E}(Y \mid A,\vec{L}) \approx \hat{g}(A,\vec{L}) + \hat{H}(A,\vec{L})\beta$$

4) Target the prediction function

$$\hat{g}'(A,\vec{L}) = \hat{g}(A,\vec{L}) + \hat{H}(A,\vec{L})\hat{\beta}$$

Targeted prediction rule optimized for the way we will aggregate predictions
Original prediction rule optimized for disaggregate prediction

5) Estimate using the targeted prediction function

$$\hat{\mathbf{E}}(Y(a)) = \frac{1}{n}\sum_{i=1}^{n}\hat{g}'(a,\vec{L}_i)$$

Fig. 9. Targeted learning with a continuous outcome. This method is analogous to the method for a binary outcome presented in Fig 8. We include the continuous version here for comparison with double machine learning (Appendix Fig 10).

1) Estimate initial prediction functions

$$\hat{g}(a,\vec{\ell}) \approx \mathbf{E}\left(Y \mid A=a, \vec{L}=\vec{\ell}\right)$$

$$\hat{m}(a,\vec{\ell}) \approx \mathrm{P}\left(A=a \mid \vec{L}\right)$$

2) Define the clever covariate

$$\hat{H}(A,\vec{L}) = \frac{\mathbb{I}(A=a)}{\hat{m}(a,\vec{L})}$$

Sample split: Carry out step 3 in a different sample from steps 1 and 2

3) De-bias the estimated outcome predictions using the clever covariate as a weight

$$\underbrace{\hat{g}'(a,\vec{\ell})}_{\text{Debiased outcome prediction}} = \underbrace{\hat{g}(a,\vec{\ell})}_{\text{Original outcome prediction}} + \underbrace{\frac{\sum_i \big(Y_i - \hat{g}(A_i,\vec{L}_i)\big) H(A_i,\vec{L}_i)}{\sum_i H(A_i,Y_i)}}_{\text{Bias correction (estimated in a \textbf{new sample})}}$$

4) Estimate using the targeted prediction function

$$\hat{\mathbf{E}}(Y(a)) = \frac{1}{n}\sum_{i=1}^{n} \hat{g}'(a,\vec{L}_i)$$

Fig. 10. Double machine learning. This figure presents double machine learning (Chernozhukov et al., 2018) using the language of targeted learning (e.g., "clever covariate," Van der Laan and Rose 2018) in order to emphasize the parallels between the two methods. For targeted learning with a continuous outcome, see Appendix Fig 9. For targeted learning with a binary outcome, see Fig 8.