

Chapter 18

VARIABLE SELECTION AND HIGH-DIMENSIONAL DATA

In the previous chapters, we have described several adjustment methods to estimate the causal effect of a treatment A on an outcome Y , including stratification and outcome regression, standardization and the parametric g-formula, IP weighting, and g-estimation. Each of these methods carry out the adjustment in different ways but all these methods rely on the same condition: the set of adjustment variables L must include sufficient information to achieve conditional exchangeability between the treated $A = 1$ and the untreated $A = 0$ —or, equivalently, to block all backdoor paths between A and Y without opening other biasing paths.

In practice, a common question is how to select the variables L for adjustment. This chapter offers some guidelines for variable selection when the goal of the data analysis is causal inference. Because the variable selection criteria for causal inference are not the same as for prediction, widespread procedures for variable selection in predictive analyses may not be directly transferable to causal analyses. This chapter summarizes the problems of incorrect variable selection in causal analyses and outlines some practical guidance.

18.1 The different goals of variable selection

As we have discussed throughout this book, valid causal inferences usually require adjustment for confounding and other biases. When an association measure between a treatment A and an outcome Y may be partly or fully explained by confounders L , adjustment for these confounders needs to be incorporated into the data analysis. Otherwise, the association measure cannot be interpreted as a causal effect measure.

Even if the outcome model includes all confounders for the effect of A on Y , the association between each confounder and the outcome cannot be causally interpreted because we do not adjust for the confounders of the confounders.

Reminder: Confounding is a causal concept that does not apply when the estimand is an association rather than a causal effect.

But if the goal of the data analysis is purely predictive, no adjustment for confounding is necessary. If we just want to quantify the association between smoking cessation A and weight gain Y , we simply estimate that association from the data by comparing the average weight gain between those who did and did not quit smoking. More generally, if we want to develop a predictive model for weight gain, we will want to include covariates (like smoking cessation, baseline weight, and annual income) that predict weight gain. We do not ask the question of whether those covariates are confounders because there is no treatment variable whose effect can be confounded. In predictive models, we do not try to endow any parameter estimates with a causal interpretation and therefore we do not try to adjust for confounding because the concept of confounding does not even apply.

The distinction between predictive/associational models and causal models was discussed in Section 15.5. Suppose clinical investigators use outcome regression to identify patients at high risk of developing heart failure. The goal is classification, which is a form of prediction. The parameters of these predictive models do not necessarily have any causal interpretation and all covariates in the model have the same status, i.e., there are no treatment variable A and adjustment variables L . For example, a prior hospitalization may be identified as a useful predictor of future heart failure, but nobody would suggest we stop admitting people to the hospital in order to prevent heart failures. Identifying

patients with bad prognosis (prediction) is different from identifying the best course of action to prevent or treat a disease (causal inference).

For pure prediction, investigators want to use variables that improve predictive ability. Most prediction algorithms include so-called tuning parameters whose values must be chosen in order to optimize predictive accuracy. For instance, the lasso and ridge regression both include a regularization parameter that shrinks regression coefficients towards zero. However the appropriate degree of shrinkage (i.e., the magnitude of the regularization parameter) needs to be adaptively chosen from the data. For neural nets the tuning parameters include the depth and width of the network. Often the choice of tuning parameter is made using cross-validation (see Fine Point 18.2). Cross-validation is also used to choose between competing algorithms as no single algorithm gives better predictions than the others in all data sets.

Because some selection algorithms such as deep neural nets are “black-box” procedures, it is not always easy to explain how the variables are selected or why the algorithms work. One point of view is that it does not necessarily matter; that is, for purely predictive purposes in a particular population and setting, whatever algorithm that works to improve prediction is fair game, regardless of interpretability.

Another point of view is that interpretable algorithms are needed because physicians will not feel comfortable in their treatment decisions, especially for patients with an unusual mixture of symptoms, if they cannot articulate to themselves and the patient the medical reasons behind these decisions. Furthermore, black-box algorithms may perform poorly when deployed to new settings because they are likely to rely on local, often noncausal, features of the training setting that are not present in the new settings. Finally, black-box algorithms can bake in often predictive, but nonetheless (socially) discriminatory, practices because the training set data were collected when those practices were in place.

A causal analysis requires different considerations. Unlike in a predictive analysis, in a causal analysis a thoughtful selection of confounders is needed if one is to believe the treatment effect estimates have a causal interpretation. Automatic variable selection procedures may work for prediction, but not generally for causal inference. Variable selection algorithms may select variables for adjustment that introduce bias in the effect estimate. There are several reasons why this bias may arise. Some of these reasons have been described earlier in the book; others have not been described yet. The next section summarizes all of them.

18.2 Variables that induce or amplify bias

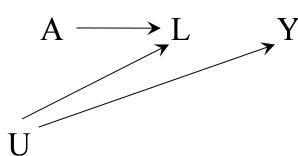


Figure 18.1

Imagine that we have unlimited computational power and a dataset with a quasi-infinite number of individuals (the rows of the dataset) and many variables measured for each individual (the columns of the data set), including treatment A , outcome Y , and a moderate number of discrete variables X , some of which may be confounders of the effect of A on Y . In this setting, we can afford to adjust for as many variables in the dataset as we wish, without computational, numerical, or statistical constraints. Thus, were our goal simply to predict Y from A and X (under a standard least squares loss; see Technical Point 18.1), we could optimally predict Y by simply using the average of Y in every joint stratum of A and X .

Fine Point 18.1

Variable selection procedures for regression models Suppose we want to fit a regression model with predictive purposes, but the database includes so many potential predictors—perhaps even more than individuals—that including all of them in the model is either impossible or results in very unstable predictions. Several approaches exist to deal with this problem in regression models. A detailed description of these procedures can be found in many books. See, e.g., the books by Hastie, Tibshirani, and Friedman (2009), and by Harrell (2015). Below we briefly outline some of the existing approaches.

One approach is to select a subset of the available variables. A conceptually simple way to find the best subset would be to first decide the number of variables in the model, then fit all possible combinations of models with that number of variables, and finally choose the best one according to some pre-specified criterion (e.g., Akaike's Information criterion). However, this approach becomes computationally infeasible for a massive number of variables and, for a finite dataset, is not guaranteed to select the model with smallest prediction error. More computationally efficient methods to select variables are forward selection (start with no variables and, in each step of the algorithm, add the variable that leads to the greatest improvement), backward elimination (start with all variables and, in each step, delete the variable that leads to the smallest improvement), and stepwise selection (a combination of forward selection and backward elimination). The variable selection algorithm ends when no further improvement is possible, with improvement again defined according to some pre-specified criterion. These algorithms are easy to implement but, on the other hand, they do not explore all possible subsets of variables.

An alternative to subset selection is shrinkage. The idea is to modify the estimation method by adding a “penalty” that forces the model parameter estimates (other than the intercept) to be closer to zero than they would have been in the absence of the penalty. That is, most parameter estimates are shrunk towards zero. As a result of this shrinkage, the variance decreases and the prediction becomes more stable. The two best known shrinkage methods are ridge regression and the *lasso* or “least absolute shrinkage and selection operator”, which was proposed by Santosa and Symes (1986) and rediscovered by Tibshirani (1996). Unlike ridge regression, the lasso allows some parameter values to be exactly zero. Therefore, the lasso is both a shrinkage method and a subset selection method.

However, suppose we want to unbiasedly estimate the average causal effect of a binary treatment A on the outcome Y , i.e., $E[Y^{a=1}] - E[Y^{a=0}]$. Then the goal of covariate adjustment is to eliminate as much confounding as possible by using the information contained in the measured variables X . We could easily adjust for all measured variables X via stratification/outcome regression, standardization/g-formula, IP weighting, or g-estimation. Are there any reasons to adjust for only a subset of X rather than simply adjust for all available variables X ? The answer is yes. Even in this ideal setting, we want to ensure that some variables are not selected for adjustment because adjustment for those variables would induce bias. The next examples illustrate this point when some of the variables L in X are causally affected by A .

Suppose the causal structure of the problem is represented by the causal diagram of Figure 18.1 (same as Figure 7.7) in which the variable L is a collider. Here the average causal effect $E[Y^{a=1}] - E[Y^{a=0}] = 0$ is unbiasedly estimated by $E[Y|A = 1] - E[Y|A = 0]$ since there is no confounding by L . Suppose now we try to estimate the average causal effect by adjusting for L via the g-formula $\sum_l E[Y|A = 1, L = l] \Pr(L = l) - \sum_l E[Y|A = 0, L = l] \Pr(L = l)$. This contrast differs from $E[Y|A = 1] - E[Y|A = 0]$ —and thus is biased—because L is both conditionally associated with Y given A and marginally associated with A , so $\Pr(L = l) \neq \Pr(L = l|A)$. Because the A - Y association adjusted for L is expected to be non-null even though the causal effect of treatment A on the outcome Y is null, we say that there is *selection bias under the null*. The same bias is expected to arise when we adjust for a variable L

Collapsibility reminder: When adjusting for covariates using stratification, remember that the adjusted association measure may differ from the unadjusted association measure, even when no confounding exists. See Fine Point 4.3.

Fine Point 18.2

Overfitting and cross-validation. Overfitting is a common problem of all variable selection methods for regression models: The variables are selected to predict the data points as well as possible, without taking in consideration that some of the variation observed in the data is purely random. As a result, the model predicts very well for the individuals used to estimate the model parameters, but the model predicts poorly for future individuals who were not used to estimate the model parameters. The same problem arises in predictive algorithms such as random forests, neural networks, and other machine learning algorithms.

A straightforward solution to the overfitting problem is to split the sample in two parts: a training sample used to run the predictive algorithm (that is, to estimate the model parameters when using regression) and a validation sample used to evaluate the accuracy of the algorithm's predictions. For a sample size n , we use v individuals for the validation set and $n - v$ individuals for the training set. When using the lasso, the degree of shrinkage in the training sample may be guided by the model's performance in the validation sample.

The obvious downside of splitting the sample into training and validation subsamples is that the predictive algorithm only uses—e.g., the model parameters are estimated in—a subset of individuals, which increases the variance. A solution is to repeat the splitting process multiple times, which increases the effective number of individuals used by the predictive algorithm. Then one can evaluate the algorithm's predictive accuracy as the average over all the validation samples. This procedure is known as *cross-validation* or out-of-sample testing. Different forms of cross-validation exist.

A procedure referred to as “leave- v -out cross-validation” analyzes all possible partitions of the sample into training sample and validation sample of size v . However, examining all such partitions may become computationally infeasible for moderately large values of n and v . Two possible fixes for this problem are (i) to choose $v = 1$ or (ii) to evaluate only a sample of the partitions. For example, in “ k -fold cross validation”, the sample is split into k subsamples of equal size. Then each one of the subsamples is used as the validation sample with the other $k - 1$ subsamples as its training sample. A common choice is $k = 10$. See the book by Hastie, Tibshirani, and Friedman (2009) for a description of cross-validation and related techniques. Deep learning algorithms based on neural networks with many layers often seem nearly immune to overfitting when massive amounts of training data are available, e.g., speech recognition, images (Goodfellow, Bengio, Courville 2016). A deep neural network is a parametric model with often thousands, millions, or even billions unknown parameters and therefore often fits the training data exactly. Astonishingly, the fitted model still can successfully predict the outcomes of future individuals (sampled from the same population) with small error. Trying to explain this phenomenon is one of the most active current research areas in machine learning.

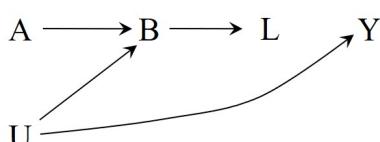


Figure 18.2

that, as in the causal diagram of Figure 18.2, is a descendant of the collider B . You may want to review Chapter 8 for more examples of causal structures with colliders and their descendants.

Selection bias may also appear when adjusting for a noncollider affected by treatment, like the variable L in the causal diagram in Figure 18.3. Here the average causal effect $E[Y^{a=1}] - E[Y^{a=0}] \neq 0$ is also unbiasedly estimated by $E[Y|A = 1] - E[Y|A = 0]$ since there is no confounding by L . However, if we try to estimate the average causal effect by adjusting for L (as if it were a pre-treatment variable), the g-formula contrast will differ from $E[Y|A = 1] - E[Y|A = 0]$ for the same reasons as in the previous paragraph.

Now suppose that the arrow from A to Y had been absent, i.e., that the null hypothesis of no effect of A on Y were true and so $E[Y^{a=1}] - E[Y^{a=0}] = 0$. Then A and Y would be independent (both marginally and conditionally on L) and the g-formula contrast would be zero and thus unbiased. The key reason for this result is that, under the null, A no longer has a causal effect on L . That is, unlike in Figures 18.1 and 18.2, adjusting for L in Figure 18.3 results in selection bias only when A has a non-null causal effect on Y . We then say that there is *selection bias under the alternative* or off the null (see Section 6.5).



Figure 18.3

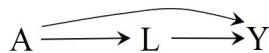


Figure 18.4

In Figure 18.4, adjusting for L blocks the path $A \rightarrow L \rightarrow Y$ but not the path $A \rightarrow Y$. Thus the $A-Y$ association adjusted for L is a biased estimator of the total effect of A on Y but an unbiased estimator of the direct effect of A on Y that is not mediated through L .

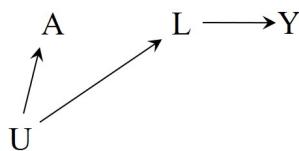


Figure 18.5

An example of the application of expert knowledge to adjustment was described by Hernán et al (2002).

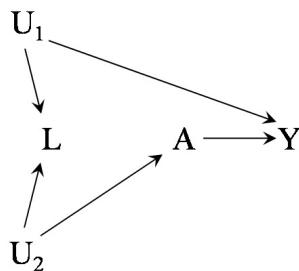


Figure 18.6

When the adjustment variable is affected by the treatment A and affects the outcome Y , we say that the variable is a *mediator*. Consider the causal diagram in Figure 18.4, which includes the mediator L on a causal path from the treatment A to the outcome Y . The $A-Y$ association adjusted for the mediator L , or its descendants, will differ from the effect of treatment A on the outcome Y because the adjustment blocks the component of the effect that goes through L . Sometimes this problem is referred to as *overadjustment for mediators* when the average causal effect of A on Y is the contrast of interest.

The bias-inducing variables discussed above share a common feature: they are affected by treatment and therefore they are post-treatment variables. One might then think that we should always avoid adjustment for variables that occur after treatment A . The rule of not adjusting for post-treatment variables would be easy to follow because the temporal sequence of the adjustment variables and the treatment is usually known. Unfortunately, following this simple rule may result in the exclusion of useful adjustment variables, as we discussed in Fine Point 7.4. Consider the causal diagram in Figure 18.5. The variable L is a post-treatment variable, but it can be used to block the backdoor path between treatment A and outcome Y . Therefore, the $A-Y$ association adjusted for L is an unbiased estimator of the effect of A on Y , whereas the unadjusted $A-Y$ association is a biased estimator. The take home message is that causal graphs do not care about temporal order. Thus, when A does not affect L , the correct analysis must be the same whether L is temporally before or temporally after A .

The problem is that, even when we know the temporal order of the variables, we cannot determine from the data whether or not A affects L . In fact, given the temporal ordering $A \ L \ Y$, any joint distribution of (A, L, Y) without any independencies is compatible with several causal graphs. So the decision whether to adjust for L must be based on information outside of the data. That is, whether to adjust for L cannot be determined via any automated procedures that rely exclusively on statistical associations. For example, as discussed in Chapter 7, there is no way to distinguish a collider from a confounder by using data only. Rather, the exclusion of bias-inducing variables from the adjustment set needs to be guided by subject-matter knowledge about the causal structure of the problem.

We next turn to the question of adjustment for variables L that are temporally prior to treatment A , i.e., our temporal ordering is now $L \ A \ Y$. Suppose, for simplicity, that the sample size is very large, greatly exceeding the number of covariates X available for adjustment. As a consequence, the variance of any estimator will be negligible and the only issue is bias. In this setting it is commonly believed that an estimator that adjusts for all available pre-treatment covariates will minimize the bias. However, this belief is wrong for two separate reasons.

Consider the causal diagram of Figure 18.6 (same as Figure 7.4), which includes a pre-treatment variable L . Because L is a collider on a path from A to Y , adjusting for it will introduce selection bias, which we referred to as M-bias in Chapter 7. Again, the observed data cannot distinguish between confounders and colliders, so one must rely on whatever external information one may have to decide whether or not to adjust for a pre-treatment variable L . In fact, it is also possible that L is both a confounder and a collider—if there were an arrow from L to A in Figure 18.6—which implies that the average causal effect cannot be identified, regardless of whether we do or do not adjust for L .

There is one additional reason to avoid indiscriminate adjustment for pre-

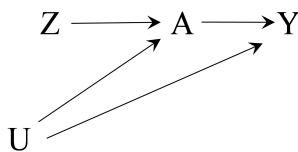


Figure 18.7

Bias amplification is guaranteed if all the equations in the structural equation model corresponding to the causal diagram are linear (Wooldridge 2010, Pearl 2011), but may also occur in more realistic settings (Ding et al. 2017).

treatment variables: *bias amplification*, a phenomenon we have not yet described in this book. Consider the causal diagram of Figure 18.7 (same as Figure 16.1), which represents a setting in which the causal effect of treatment A on the outcome Y is confounded by the unmeasured variable U . Because U is not available in the data, we cannot adjust for U and the confounding is intractable. Adjustment for the variable Z —using the g-formula as above with L replaced by Z —does not eliminate confounding because Z is not on any backdoor path from the treatment A to the outcome Y . In fact, Z is an instrument—which can be used for instrumental variable estimation in some situations described in Chapter 16—and therefore useless for direct confounding adjustment by the g-formula.

Interestingly, even though Z cannot be used to adjust away the confounding bias due to U , adjustment for the instrument Z can amplify the confounding bias due to U . That is, the A - Y association adjusted for Z may be further from the effect of A on Y than the A - Y association not adjusted for Z . This bias amplification due to adjusting for an instrument Z , often referred to as Z -bias, is a reason to avoid adjustment for variables that, like Z , are instruments. Bias amplification, however, is not guaranteed: adjustment for Z could also reduce the bias due to confounding by the unmeasured variable U . Generally, it is not possible to know whether adjustment for an instrument will amplify or reduce bias.

In summary, even if we had no computational constraints and a quasi-infinite sample size, it is not advisable to adjust for all available variables X . Ideally, the adjustment set would not include any variables that may introduce or amplify bias. Because these bias-inducing variables cannot be empirically identified by purely statistical algorithms, expert knowledge is needed to guide variable selection.

18.3 Causal inference and machine learning

For the remainder of this chapter, we will assume that we have somehow succeeded at ensuring that X includes no variables that may induce or amplify bias (i.e., no variables that would destroy conditional exchangeability if adjusted for) while still including all confounders L of the average causal effect of A on Y (i.e., all variables needed to achieve conditional exchangeability). Furthermore, we assume positivity holds. Our next problem is to estimate this effect $E[Y^{a=1}] - E[Y^{a=0}]$ in practice when X is very high-dimensional or includes multiple continuous variables.

If we have good estimates of $E[Y^{a=1}]$ and $E[Y^{a=0}]$, their difference will be a good estimate of $E[Y^{a=1}] - E[Y^{a=0}]$. Thus, for simplicity, we will focus on the estimation of $E[Y^{a=1}]$.

Depending on the adjustment method that we choose, the variables X will be used in different ways. When using the plug-in g-formula (standardization) to estimate $E[Y^{a=1}]$, we will estimate the mean outcome Y conditional on the variables X among individuals with $A = 1$, which we refer to as $b(X)$; when using IP weighting, we will estimate the probability of treatment A conditional on the variables X , which we refer to as $\pi(X)$. We can produce estimates $\hat{b}(x)$ and $\hat{\pi}(x)$ via the sort of traditional parametric models (e.g., generalized linear models with linear, logistic, or log links) with the number of parameters much smaller than the sample size that we have described in Part II of this book. When X is high-dimensional, such models are certain to be misspecified. As

a consequence, both $\hat{b}(x)$ and $\hat{\pi}(x)$ will fail to be consistent for the true $b(x)$ and $\pi(x)$.

To reduce the possibility of model misspecification, we might want to fit richly parameterized generalized linear models with linear predictor $\theta^T s(x) = \sum_j \theta_j s_j(x)$, where $s(X)$ is a very high-dimensional vector of transformations of the covariate vector X . The vector $s(X)$ generally contains both flexible high-dimensional transformations (e.g., cubic splines) of individual variables in X and cross-variable products of these transformations. For concreteness, suppose we choose a logit link so $\hat{b}(x) = \text{expit}(\hat{\theta}_b^T s(x))$ and $\hat{\pi}(x) = \text{expit}(\hat{\theta}_\pi^T s(x))$ where θ_b and θ_π are the parameters of the models for $b(x)$ and $\pi(x)$. Even if the estimated functions $\hat{b}(x)$ and $\hat{\pi}(x)$ based on these models are consistent, in finite samples the errors $\hat{b}(x) - b(x)$ and $\hat{\pi}(x) - \pi(x)$ will be much greater than would be the case for a *correctly specified* low-dimensional parametric model. In fact the dimension of $s(X)$ may frequently exceed the number of individuals n contributing data to the study. In that case, a fit of the model will fail to converge and no estimate of θ will be returned.

Possible ways forward are to fit the parametric model with linear predictor $\theta^T s(X)$ by adding a lasso or ridge penalty (see Fine Point 18.1), to use a variable selection algorithm such as stepwise selection, or to estimate the conditional expectations $b(X)$ and $\pi(X)$ using other predictive machine learning algorithms such as tree-based algorithms (e.g., random forests) or neural networks (e.g., deep learning). As discussed in Fine Point 18.2, deep learning algorithms fit a model often containing thousands or millions of parameters. Other machine learning algorithms also effectively fit thousands of parameters. In most cases with large sample sizes and many covariates X , machine learning algorithms outperform traditional parametric models for the accurate prediction of conditional expectations.

However, predictive machine learning algorithms do not by themselves suffice to adequately adjust for confounding in high-dimensional settings. In the next section we explain that these algorithms must be used in conjunction with doubly robust estimators with two modifications: sample splitting and cross-fitting. This is necessary if we hope to construct valid 95% Wald confidence intervals, i.e., intervals that trap the causal parameter of interest at least 95% of the time.

Remember that some of the variables in X may not even be confounders so we would not need to adjust for them if we knew which variables they were.

Machine learning algorithms can use cross-validation (see Fine Point 18.2) to optimize predictive accuracy.

18.4 Doubly robust machine learning estimators

Valid Wald intervals for $\psi = E[Y^{a=1}]$ require that the bias of the estimator be much less than the standard error of the estimator. The standard error of most estimators $\hat{E}[Y^{a=1}]$ of $E[Y^{a=1}]$ scale as $1/\sqrt{n}$ times a constant, where n is the sample size. Hence, we require that the bias of $\hat{E}[Y^{a=1}]$ to be much less than $1/\sqrt{n}$. In addition to small bias, in order to have valid Wald intervals centered on $\hat{E}[Y^{a=1}]$, we generally need $\hat{E}[Y^{a=1}]$ to also be asymptotically normal, which is generally easier to achieve than small bias.

A small bias is easier to achieve with doubly robust estimators than with non-doubly robust estimators, because the bias $\hat{E}[Y^{a=1}] - E[Y^{a=1}]$ of a doubly robust estimator depends on the product of the errors $\frac{1}{\pi(x)} - \frac{1}{\hat{\pi}(x)}$ and $b(x) - \hat{b}(x)$, which can be small. Indeed the bias is less than $1/\sqrt{n}$ if both errors are much smaller than $1/\sqrt{n}$, which can often be achieved by machine

The degree of undercoverage will be greater when there is some degree of confounding in the super-population since, in that case, Wald confidence intervals will not be centered on an unbiased estimator of the causal effect (see Chapter 10).

This property of doubly robust estimators is referred to as a *second-order bias*. See Technical Point 13.2 for details.

We may refer to the training sample as the *nuisance sample* because we use it to estimate the nuisance regressions for $b(X)$ and $\pi(X)$. Fine Point 15.1 reviews the concept of nuisance parameters.

Sample splitting and cross-fitting are not new procedures. However, the idea of combining these procedures with machine learning has not been emphasized until recently.

learning estimators if the functions $\pi(x)$ and $b(x)$ are either quite smooth, i.e., have many derivatives, or very sparse, i.e., depend on only few components of the vector X , even though how many and which components are unknown. In contrast, the IP weighted and plug-in g-formula estimators of $E[Y^{a=1}]$ can have bias as large as the errors $\frac{1}{\pi(x)} - \frac{1}{\hat{\pi}(x)}$ and $b(x) - \hat{b}(x)$, respectively. If so, neither the IP weighted estimator nor the plug-in g-formula estimator of $E[Y^{a=1}]$ can generally center a valid Wald confidence interval because, with high-dimensional X , it is known that the error of any possible estimator $\hat{\pi}(x)$ or $\hat{b}(x)$ of $\pi(x)$ or $b(x)$ must exceed $1/\sqrt{n}$.

But, if we hope to construct valid 95% Wald confidence intervals, the *doubly robust machine learning estimators* of the previous paragraph need to incorporate sample splitting and cross-fitting. We now describe these two procedures and their rationale. Technical Point 18.1 summarizes the steps of the estimation process.

We begin by describing *sample splitting*. First, we randomly divide the study population of n individuals into two halves: an estimation sample of size $n/2$ and a training sample of equal size. Second, we apply the predictive machine learning algorithms to the training sample in order to obtain estimators of $\hat{b}(x)$ and $\hat{\pi}(x)$ for the conditional expectations $b(x) = E[Y|X = x, A = 1]$ and $\pi(x) = E[A|X = x]$, respectively. Third, we compute the doubly robust estimator of the average causal effect in the estimation sample using the estimators of $\hat{b}(x)$ and $\hat{\pi}(x)$ from the training sample. We have now obtained a doubly robust machine learning estimate of the average causal effect in a random half of the study population.

To understand the need for sample splitting, let us compare the split-sample version with the full-sample version of the augmented IP weighted (AIPW) doubly robust estimator of Technical Point 13.2. The estimation sample used in the split-sample AIPW estimator of $E[Y^{a=1}]$ is statistically independent of the split-sample estimators of $\hat{b}(x)$ and $\hat{\pi}(x)$, which use only the training sample data. As a consequence, under weak conditions described in Technical Point 18.2, the estimator is asymptotically normal with standard error that scales like $1/\sqrt{n}$ with the product bias described earlier. It follows that, if the product bias is less than $1/\sqrt{n}$, Wald intervals centered on the split-sample estimator will be valid.

In contrast, the full-sample AIPW estimator of $E[Y^{a=1}]$ is

$$\frac{1}{n} \sum_{i=1}^n \left[\hat{b}(X_i) + \frac{A_i}{\hat{\pi}(X_i)} \{Y_i - \hat{b}(X_i)\} \right],$$

where $\hat{b}(x)$ and $\hat{\pi}(x)$ are now estimated by a machine learning algorithm applied to all n individuals' data. Thus $\hat{b}(x)$ and $\hat{\pi}(x)$ are correlated with the full-sample AIPW estimator. This correlation, if sufficiently large, can affect the bias, variance, and asymptotic normality of the full-sample estimator in unpredictable ways. Unfortunately, the magnitude of the correlation is unknown and cannot be well estimated. Hence, the split-sample estimator is much preferred to the full-sample estimator in high-dimensional settings.

The only difficulty with using the doubly robust split-sample estimator is that its variance and standard error correspond to a sample size of $n/2$. As a result, our confidence interval will be wider than the one we would have obtained if we had been able to use the entire sample of n individuals. A way to overcome this problem is cross-fitting.

We now describe how *cross-fitting* recovers the statistical efficiency lost by sample splitting. First, we repeat the above procedure but swapping the roles

Technical Point 18.1

Augmented IP weighted split-sample and cross-fit estimator. The augmented IP weighted (AIPW) estimator of $\psi = E[Y^{a=1}]$ is a doubly robust estimator described in Technical Point 13.2. The following algorithm computes the AIPW split-sample estimator $\widehat{\psi}$ and the cross-fit estimator $\widehat{\psi}_{\text{cross-fit}}$:

- (i) Randomly split the n study subjects into 2 parts: an *estimation* sample of size q and a *training* sample of size $n_{\text{tr}} = n - q$ with $q/n \approx 1/2$.
- (ii) Estimate $\widehat{b}(x)$ and $\widehat{\pi}(x)$ of $b(x) = E[Y|A=1, X=x]$ and $\pi(x) = pr[A=1|X=x]$ from the training sample data using machine learning algorithms.
- (iii) Compute the split-sample AIPW estimator

$$\widehat{\psi} = \frac{1}{q} \sum_{i=1}^q \left[\widehat{b}(X_i) + \frac{A_i}{\widehat{\pi}(X_i)} \{Y_i - \widehat{b}(X_i)\} \right]$$

from the q subjects in the estimation sample.

- (iv) Compute the cross-fit estimator

$$\widehat{\psi}_{\text{cross-fit}} = (\widehat{\psi} + \overline{\widehat{\psi}})/2$$

where $\overline{\widehat{\psi}}$ is $\widehat{\psi}$ but with the training and estimation sample swapped.

An alternative cross-fit estimator with improved finite sample behavior is computed as follows: (i) divide the sample of size n into $M > 2$ equal-sized random samples, (ii) compute $\widehat{\psi}^{(m)}$, $m = 1, 2, \dots, M$, using sample m as estimation sample and the remaining $M - 1$ samples as the training sample, and (iii) compute $\widehat{\psi}_{\text{cross-fit}} = \frac{1}{M} \sum_{m=1}^M \widehat{\psi}^{(m)}$.

of the estimation and training halves of the study population. That is, we use the half formerly reserved for estimation as the new training sample, and the half formerly used for training as the new estimation sample. We then compute the doubly robust estimator of the average causal effect in the new estimation sample using the estimators of $\widehat{b}(x)$ and $\widehat{\pi}(x)$ from the new training sample. We have now obtained a doubly robust machine learning estimate of the average causal effect in the other random half of the population.

The next step is to compute the average of the two doubly robust estimates from each half of the population. This average will be our doubly robust estimate of the effect in the entire study population. A 95% confidence interval around this estimate can be constructed by bootstrapping, either by adding and subtracting 1.96 times the bootstrap standard error or by using the 2.5 and 97.5 percentiles of the bootstrap estimates as the bounds of the interval.

We are done. Through sample splitting and cross-fitting, we can combine doubly robust estimation and machine learning to obtain causal effect estimates which have known statistical properties and which use all the available data. An active area of research is the development of procedures to detect whether the bias of doubly robust split-sample estimators is the order of or larger than the standard error and, if so, to obtain estimates with smaller bias in the estimation sample without having to redo the machine learning component in the training sample.

Lin et al. (2020) constructed estimators based on higher order influence functions that had smaller bias than doubly robust cross-fit estimators without significantly increasing their variance.

Technical Point 18.2

Statistical properties of split-sample and cross-fit estimators. Conditional on the training sample data T_r , $\hat{b}(x)$ and $\hat{\pi}(x)$ are fixed functions. Hence $\hat{\psi}$ is the sum of independent and identically distributed random variables and thus, by the central limit theorem, it is asymptotically normal conditional on T_r with standard error $se(\hat{\psi})$ proportional to $n^{-1/2}$. The exact conditional bias of $\hat{\psi}$ is

$$E[\hat{\psi} - \psi | T_r] = E\left[\pi(X_i) \left(\frac{1}{\hat{\pi}(X_i)} - \frac{1}{\pi(X_i)}\right) \{b(X) - \hat{b}(X)\} | T_r\right]$$

To characterize the unconditional statistical properties of $\hat{\psi}$ and $\hat{\psi}_{\text{cross-fit}}$, we must take into account that $E[\hat{\psi} - \psi | T_r]$ is random through its dependence on the training sample data via \hat{b} and $\hat{\pi}$. If (i) $\hat{b}(x)$ and $\hat{\pi}(x)$ are consistent for the true $b(x)$ and $\pi(x)$ (in mean square), and (ii) $E[\hat{\psi} - \psi | T_r] / se(\hat{\psi})$ converges to 0 in probability, then $\hat{\psi}$ and $\hat{\psi}_{\text{cross-fit}}$ are asymptotically normal and unbiased.

Thus, when (i) and (ii) hold, 95% Wald confidence intervals $\hat{\psi} \pm 1.96 \times \hat{se}(\hat{\psi})$ and $\hat{\psi}_{\text{cross-fit}} \pm 1.96 \times \hat{se}(\hat{\psi}_{\text{cross-fit}})$ are valid and, in fact, are calibrated. Here $\hat{se}(\hat{\psi}_{\text{cross-fit}})$ and $\hat{se}(\hat{\psi})$ can be computed with the bootstrap. Further, $n^{1/2} \hat{se}(\hat{\psi}_{\text{cross-fit}})$ is semiparametric efficient with standard error $\left\{var\left\{b(X) + \frac{A}{\pi(X)} [Y - b(X)]\right\}\right\}^{1/2}$, which is smaller than the standard error of $\hat{\psi}$ by a factor of $1/\sqrt{2}$. Note if the rate of convergence of $\frac{1}{\hat{\pi}(x)} - \frac{1}{\pi(x)}$ is $n^{-\alpha}$ and that of $b(x) - \hat{b}(x)$ is $n^{-\epsilon}$, the bias $E[\hat{\psi} - \psi | T_r]$ is $o(n^{-1/2})$ if $\alpha + \epsilon > 1/2$. Thus if $\hat{b}(x)$ has a rate of convergence slower than $n^{-1/4}$, the bias can still be $o(n^{-1/2})$ if $\hat{\pi}(x)$ has a sufficiently fast rate of convergence. The same holds with the roles of $\hat{b}(x)$ and $\hat{\pi}(x)$ swapped.

18.5 Variable selection is a difficult problem

The methods outlined in the previous section invalidate the widespread belief that any data-adaptive procedure to select adjustment variables will inevitably result in incorrect confidence intervals. As we have seen, the combination of causal inference methods with machine learning algorithms for confounder selection can, under certain conditions, result in correct statistical inference. However, doubly robust machine learning does not solve all our problems for at least three reasons (in addition to that described in the previous section).

First, in many applications, the available subject-matter knowledge may be insufficient to identify all important confounders or to rule out variables that induce or amplify bias. Thus there is no guarantee that doubly robust machine learning estimators will have a small bias.

Second, the implementation of doubly robust estimators has been difficult—and computationally expensive when combined with machine learning—in high-dimensional settings with time-varying treatments. This is especially true for causal survival analysis. As a result, most published examples of causal inference from complex longitudinal data use single robust estimators, which are the ones we have largely emphasized in Part III of this book. However, the methods outlined in this chapter are quickly becoming routine in some fields.

Third, doubly robust machine learning can yield a variance of the causal effect that equals the variance that would have been obtained if the true conditional expectations $b(X)$ and $\pi(X)$ were known. However, there is no guarantee that such variance will be small enough for meaningful causal inference.

Suppose that we obtain a doubly robust machine learning estimate of the causal effect, as described in the previous section, only to find out that its (correct) variance is too big to be useful. This will happen, even when we have estimated the propensity score and outcome regression with small product bias, if some of the covariates in X are strongly associated with the treatment A . Then the probability of treatment $\pi(X)$ may be near 0 or near 1 for individuals with a particular value of X . As a result, the effect estimate will have a very large variance and thus a very wide (but often correct) 95% confidence interval. Since we do not like very wide 95% confidence intervals, even if they are correct, we may be tempted to throw out the variables in X that are causing the “problem” and then repeat the data analysis. If we did that, we would be fundamentally changing the game. Using the data to discard covariates in X that are associated with treatment, but not so much with the outcome, makes it no longer possible to guarantee that the 95% confidence interval around the effect estimate is valid. The tension between including all potential confounders to eliminate bias and excluding some variables to reduce the variance is hard to resolve.

Given all of the above, developing a clear set of general guidelines for variable selection may not be possible. In fact, so much methodological research is ongoing around these issues that this chapter cannot possibly be prescriptive. As discussed in Section 13.5, the best scientific advice for causal inference may be to carry out multiple sensitivity analyses: implement several analytic methods and inspect the resulting effect estimates. If the various effect estimates are compatible, we will be more confident in the results. If the various effect estimates are not compatible, our job as researchers is to try to understand why.

This result raises a puzzling philosophical question: If the confidence interval is invalid when we use the data to rule out, say, 5 variables that make the variance too large, then why should the confidence interval be valid if we had happened to receive a dataset that did not include those 5 variables? Given that we always work with datasets in which some potential confounders are not recorded, how should we interpret confidence intervals in any observational analysis?

Part III

Causal inference for time-varying treatments

