Machine Learning and Causal Inference.

Causal Inference.

Eduardo Fé

05/12/2023

Machine Learning and Causal Inference

Using Machine Learning to estimate causal effects. . .

. . . is complicated. . . .

Setting

We are going to avoid the potential outcomes framework.

Instead, we will think *à la econometrique*, from a regression perspective.

- Outcome, Y
- Some nuisance predictors, X_j, $j = 1, ..., p$
- A binary treatment T ($T = 1$ if a person receives the treatment, 0 otherwise)
- And we might also have an *instrumental variable*, Z
- An *i.i.d.* error term, ε_i

In general,

$$Y_i = \tau \cdot T_i + g(X_{1,i}, ..., X_{p,i}) + \varepsilon_i \tag{1}$$

Thus X_j and Y are associated, but we do not care about the specific form of this relationship (a *nuisance relationship*).

Setting

The parameter of interest is τ, the Treatment/Causal Effect of T.

We will simplify the discussion by assuming:

- The treatment is one, and always the same for everybody.
- A person's treatment uptake does not affect a different person's level of Y.
- The treatment effect τ is constant.

We have a cross-section of data, $(Y_i, T_i, X_{1,i}, ..., X_{p,i})$.

We will define the relationships between T, X, Z and ε as we go along.

Regularisation (again)

Shrinkage estimators minimise the RSS but they introduce a penalty as model complexity increases.

$$\text{argmin}_{\beta_j} \sum_{i=1}^{n}(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j \cdot X_{j,i})^2 + \lambda \cdot \text{penalty}(\beta_1, ..., \beta_p)$$

Here $\lambda \geq 0$ is chosen by the researcher (potentially via a data-driven method). The larger λ, the larger complex models are penalised.

The idea was to reduce the variance in our predictions (through the penalty) at the expense of introducing some bias (in our estimates of the coefficients, β_j).

Regularisation (again).

We simulated data from the following model:

$$Y_i = \sum_{j=1}^{100} X_{j,i} \cdot \beta_j + v_i$$

- $X_{j,i}$ have a multivariate normal distribution, with correlation between $X_{j,i}, X_{k,i}$ equalt to 0.1.
- v_i follows a standard normal
- The first ten β_j take values 2.000, 1.789 ,1.578, 1.367, 1.156, 0.944 ,0.733, 0.522 ,0.311, 0.100
- The remaining 90 β_j take on a small value (between 0.001 and 0.005)

We created samples of 500 observations 500 times. We `set.seed(123)`.

Regularisation (again).

We first focus on the estimation of the largest parameter, $\beta_1 = 2$.

Mean (OLS)	2.00
Mean(LASSO)	1.96
Mean Bias, (OLS)	0.00
Mean Bias, (LASSO)	-0.04
Mean Absolute Error (OLS)	0.04
Mean Absolute Error (LASSO)	0.05
Dropped from the model (LASSO)	0

Regularisation (again).

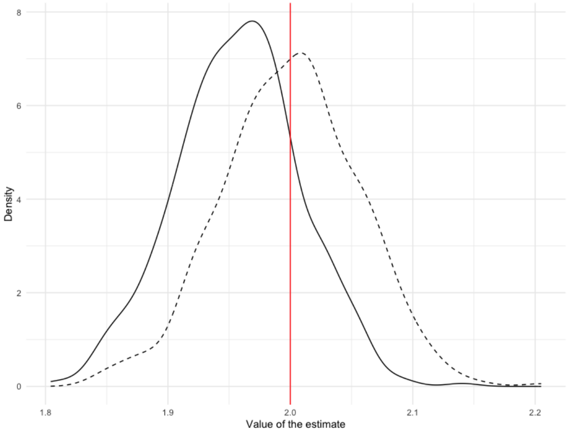

Figure 1: Distribution of estimates of the coefficient for X1. The true value of the parameter was 2, and is marked with a red vertical line. The distribution of OLS corresponds to the dashed curve; the distribution of LASSO estimates corresponds to the solid curve.

Regularisation (again).

We now focus on the estimation of the smallest parameter, $\beta_{10} = 0.1$.

Mean (OLS)	0.096
Mean(LASSO)	0.0587
Mean Bias, (OLS)	0.0866
Mean Bias, (LASSO)	0.0487
Mean Absolute Error (OLS)	0.0882
Mean Absolute Error (LASSO)	0.0514
Dropped from the model (LASSO)	10%

Regularisation (again).

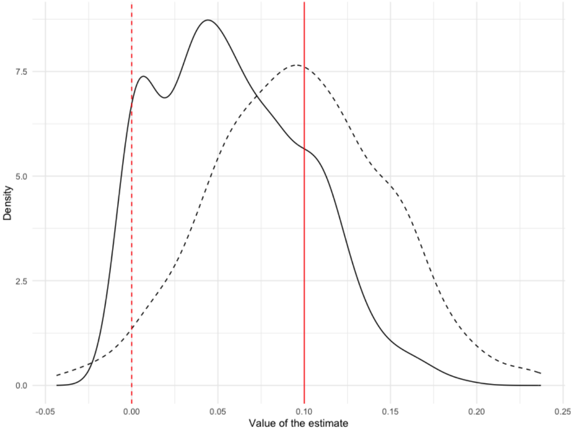

Figure 2: Distribution of estimates of the coefficient for X10. The true value of the parameter was 0.1, and is marked with a red vertical line. The dashed vertical line mark the value 0. The distribution of OLS corresponds to the dashed curve; the distribution of LASSO estimates corresponds to the solid curve. The hump at 0 simply reflects that in a substantive proportion the simulations the coefficient of X10 was dropped by LASSO.

A direct attack.

Given the model

$$Y_i = \tau \cdot T_i + g(X_{1,i}, ..., X_{p,i}) + \varepsilon_i \tag{2}$$

One could try to estimate τ directly:

- Use a Machine Learning method. For instance, we could use Random Forest (or LASSO with many powers and interactions of the $X_{j,i}$) to approximate $g(.)$.
- You could run a bootstrap, to obtain standard errors for the estimate of τ.

However the previous discussion already suggests that this will lead to trouble.

A direct attack.

Specifically, in the previous simulation we could assume that

$$
\begin{aligned}
Y_i &= \tau \cdot T_i + g(X_{1,i}, ..., X_{p,i}) + \varepsilon_i \\
&= \sum_{j=1}^{100} X_{j,i} \cdot \beta_j + v_i
\end{aligned} \tag{3}
$$

Where $\tau = \beta_1$ (or $\tau = \beta_{10}$) and $g(.)$ is represented by the remaining linearly additive terms.

The setting would be one of treatment estimation under "**unconfounded**" assignment conditional on observables (or under the **Conditional Independence Assumption**).

The simulation is designed to be favourable to OLS.

But LASSO fails, particularly when the value of τ is small (as is often the case in practice).

Post-LASSO estimation.

If the problem is the bias introduced by LASSO… Can we

- Estimate a LASSO model
- Record the variables that are NOT dropped by the model
- Run OLS with the variables selected by LASSO only?

If you are doing prediction, yes…

… but you are doing Causal Inference… so, **NO**

Post-LASSO estimation

The problem of this approach is that:

- First, model selection mistakes occur (LASSO is not infallible in that domain).
- Some of the variables eliminated by LASSO might be relevant to explain treatment uptake
- Your model will fail to take into account that correlation with the deleted variables
- That will introduce a bias in the OLS estimator.

What if treatment is assigned at random?

$$Y_i = \tau \cdot T_i + g(X_{1,i}, ..., X_{p,i}) + \varepsilon_i \tag{4}$$

What if T_i is allocated at random, as in a Randomized Control Trial?

When you studied linear regression, they explained to you that, when estimating τ by OLS, the omission of variables $X_{j,i}$ is not a problem **IF** T_i is randomly allocated.

How would Machine Learning operate in that setting?

What if treatment is assigned at random?

We simulated data from the following model:

$$Y_i = \tau \cdot T_i + \sum_{j=1}^{100} X_{j,i} \cdot \beta_j + v_i$$

- $X_{j,i}$ have a multivariate normal distribution, with correlation between $X_{j,i}, X_{k,i}$ or 0.1.
- v_i follows a standard normal
- The first ten β_j take values 2.000, 1.789 ,1.578, 1.367, 1.156, 0.944 ,0.733, 0.522 ,0.311, 0.100
- The remaining 90 β_j take on a small value (between 0.001 and 0.005)
- **NOW** T_i was randomly allocated, and independently of the X's, and $\tau = 1$

We created samples of 500 observations 500 times. We `set.seed(123)`.

What if treatment is assigned at random?

Nope…

Mean (OLS)	1.01
Mean(LASSO)	0.89
Mean Bias, (OLS)	0.0053
Mean Bias, (LASSO)	-0.11
Mean Absolute Error (OLS)	0.0839
Mean Absolute Error (LASSO)	0.122
Dropped from the model (LASSO)	0%

What if treatment is assigned at random?

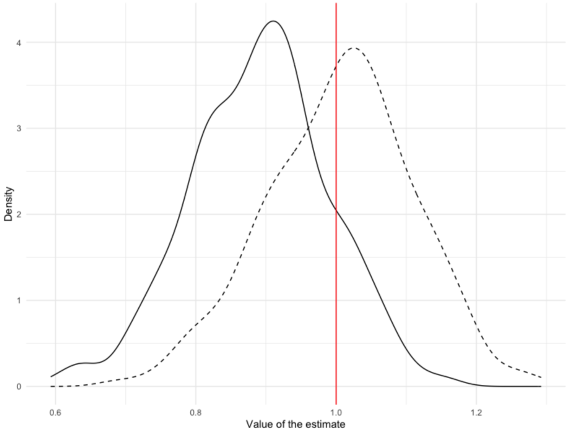

Figure 3: Distribution of estimates of the coefficient for tau when T is randomly allocated. The true value of the parameter was 1, and is marked with a red vertical line. The distribution of OLS corresponds to the dashed curve; the distribution of LASSO estimates corresponds to the solid curve.

However…

Let's keep the assumption that treatment is randomly allocated.

If the problem is regularisation, what if we leave the parameter of interest outside the regularisation? In LASSO,

$$\text{argmin}_{\beta_j} \sum_{i=1}^{n} (Y_i - \tau \cdot T_i - \beta_0 - \sum_{j=1}^{p} \beta_j \cdot X_{j,i})^2 + \lambda \cdot \sum_{j=1}^{p} |\beta_j|$$

where the penalty now does not depend on T_i.

However. . .

We run the same simulation, with T randomly allocated AND we forced the T out of the regularisation in the LASSO.

To emphasise the message, we set $\tau = 0.1$ (a small effect):

Mean (OLS)	0.0931
Mean(LASSO)	0.0961
Mean Bias, (OLS)	-0.00688
Mean Bias, (LASSO)	-0.00393
Mean Absolute Error (OLS)	0.0796
Mean Absolute Error (LASSO)	0.0726
Dropped from the model (LASSO)	0%

Not only LASSO did well now; it did better than OLS.

What if treatment is assigned at random?

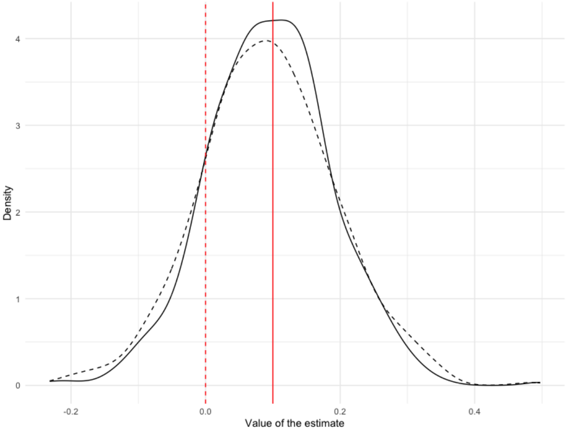

Figure 4: Distribution of estimates of the coefficient for tau, when T is randomly allocated and left out of the regularisation. The true value of the parameter was 0.1, and is marked with a red vertical line. The distribution of OLS corresponds to the dashed curve; the distribution of LASSO estimates corresponds to the solid curve.

What if treatment is assigned at random?

What is going on?

- We are applying Machine Learning to the nuisance variables
- Machine Learning does well at PREDICTING the effect of those variables.
- It does this particularly well because the X's and T are orthogonal (by randomization) so the Machine Learning method is not missing any relevant information for the predictive task.
- Then we estimate τ having discounted the variation in Y caused by the nuisance variables.
- After discounting/predicting the effect of the X's, all that remains is the variation of Y with T, and the LASSO can now pick that outstanding variation without bias (since we do not apply regularisation to T).
- Again this is possible because the X's and T are independent.

Key Insight

In essence, any causal inference problem will need to be set up in terms of

- the causal parameter of interest
- and a nuisance function, depending of other variables, that is *independent* (orthogonal really) of T, and which can be estimated by Machine Learning method.

Robinson's Semiparametric Estimator.

Suppose, again,

$$Y_i = \tau \cdot T_i + g(X_{1,i}, ..., X_{p,i}) + \varepsilon_i$$

where the treatment effect, τ is constant.

The treatment is not randomly assigned, and is correlated with the X's.

We can measure all the relevant covariates to explain Y (no ommited variables; unconfounded treatment conditional on the observables or Conditional Independence Assumption, CIA).

Estimation of τ in this setting has a long tradition in Econometrics.

Peter Robinson (1988, Econometrica), tackled this problem by using a nonparametric regression to estimate $g(.)$.

Robinson's Semiparametric Estimator.

The key insight in Robinson's paper is that,

$$\begin{aligned} E(Y_i|X_{1,i}...X_{p,i}) &= \tau \cdot E(T_i|X_{1,i}...X_{p,i}) + g(X_{1,i},...,X_{p,i}) \\ m(\mathbf{X}) &= \tau \cdot e(\mathbf{X}) + g(\mathbf{X}) \end{aligned} \tag{5}$$

because $E(\varepsilon_i|X_{1,i}...X_{p,i}) = 0$ under CIA. Above $\mathbf{X}$ stands for $X_{1,i}...X_{p,i}$.

Then, we can de-mean Y_i,

$$Y_i - m(\mathbf{X}) = \tau \cdot (T_i - e(\mathbf{X})) + \varepsilon_i \tag{6}$$

This is actually just a linear regression model **without an intercept**.

Robinson's Semiparametric Estimator.

The key insight in Robinson's paper is that,

$$Y_i - m(\mathbf{X}) = \tau \cdot (T_i - e(\mathbf{X})) + \varepsilon_i \tag{7}$$

This is actually just a linear regression model **without an intercept**.

The model thus separates the problem of inference (about τ) from the problem of regularisation:

- We can estimate $m(.), e(.)$ using *any* Machine Learning method
- We can then plug in the predictions of those models $\hat{m}(.)$, $\hat{e}(.)$ to create the *residuals*, $Y_i - \hat{m}(.)$, $T_i - \hat{e}(.)$.
- Then, we simply apply OLS on a regression of $Y_i - \hat{m}(.)$ on $T_i - \hat{e}(.)$ (without an intercept).

Robinson's Semiparametric Estimator.

We simulated data, 500 times, from the following model:

$$
\begin{aligned}
Y_i &= \tau \cdot T_i + \sin(1.5 \times X) + u_i & (8)\\
T_i &= \text{Binomial}(\pi(X_i)) & (9)\\
\pi(X_i) &= (1 + \exp(-0.5 \times X_i))^{-1} & (10)\\
X_i &\sim N(0, 1) & (11)\\
u_i &\sim N(0, 1) & (12)\\
\tau &= 2 & (13)
\end{aligned}
$$

Treatment is not randomly allocated, because it depends on X. The probability of treatment follows a logistic shape. The outcome Y depends in a nonlinear fashion on X.

Robinson's Semiparametric Estimator.

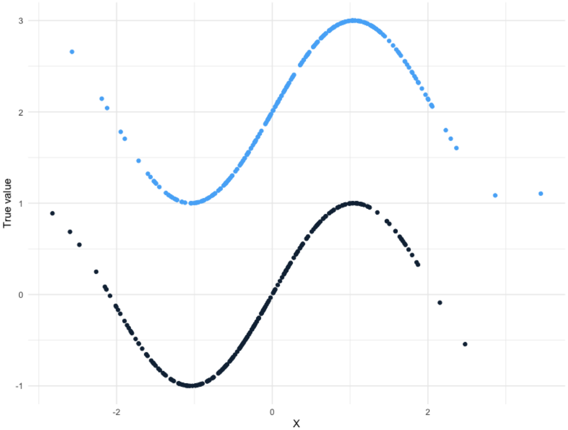

Figure 5: The conditional mean of Y given X for the treated (light blue) and control (dark blue) groups.

Robinson's Semiparametric Estimator.

We simulated data, 500 times, from the following model:

$$Y_i = \tau \cdot T_i + \sin(1.5 \times X) + u_i \tag{14}$$

$$T_i = \text{Binomial}(\pi(X_i)) \tag{15}$$

$$\pi(X_i) = (1 + \exp(-0.5 \times X_i))^{-1} \tag{16}$$

$$X_i \sim N(0, 1) \tag{17}$$

$$u_i \sim N(0, 1) \tag{18}$$

$$\tau = 2 \tag{19}$$

We can use *any* machine learning method to estimate $m(.)$ and $e(.)$. In the simulation we used LASSO:

$$E(e(X_i)|X_i) = \beta_0 + \sum_{j=1}^{5} \gamma_j \cdot X_i^j + r_{e,i} \tag{20}$$

where $r_{e,i}$ is an approximation error that vanishes at the *right* rate.

Robinson’s Semiparametric Estimator.

Mean (OLS)	2.01
Mean (Robinson)	2.00
Mean Bias, (OLS)	0.0149
Mean Bias, (Robinson)	-0.00471
Mean Absolute Error (OLS)	0.0872
Mean Absolute Error (Robinson)	0.0786

Robinson’s method gives a 10% improvement in Mean Absolute Error.

Robinson’s Semiparametric Estimator.

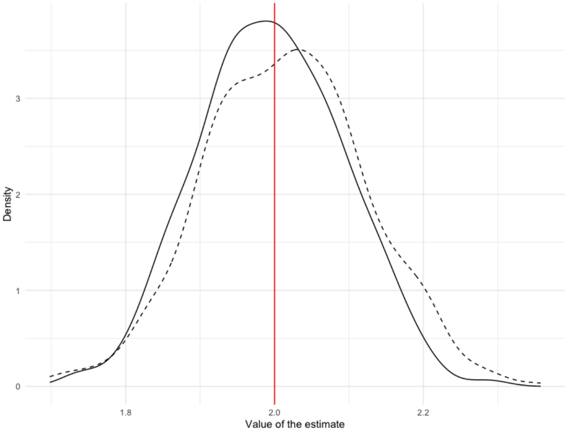

Figure 6: Distribution of estimates of the coefficient for tau, when T is not randomly allocated (OLS v Robinson’s Semiparametric Estimator). The true value of the parameter was 2, and is marked with a red vertical line. The distribution of OLS corresponds to the dashed curve; the distribution of Robinson’s estimates corresponds to the solid curve.

Robinson's Semiparametric Estimator.

Interestingly, the standard errors produced by OLS in the last stage of the estimation method are **valid** (up to the usual disclaimers -heteroskedasticy, clustered standard errors, validity of CIA, etc)

This means we can do hypothesis testing on the parameter and compute confidence intervals in the usual fashion

Conclusion

- Machine Learning methods can do a good job at predicting/classifying outcomes.
- However they cannot be used, out of the box, to estimate causal parameters.
- The key to combining ML and Causal Inference is to find methods that “orthogonalise” the prediction and inference parts of any given problem.
- We have seen that in RCT we can get good estimates by excluding the treatment parameter from the regularisation part.
- Elsewhere, things are more complex
- Under Conditional Independence/Unconfounded treatment conditional on observables, we can apply Robinson’s Semiparametric method to get a **CONSTANT** treatment effect.
- Things get more complicated as soon as we start to consider Heterogeneous (not constant) causal effects or we move away from the RCT/Unconfounded/IV framework.

Conclusion.

There are loads of resources out there (mostly in the form of published papers)

- Susan Athey, Guido Imbens and Stefan Wager (work on heterogeneous causal effects, among others).
- Victor Chernozhukov, Alexandre Belloni and Christian Hanse (and colleagues) are behind the idea of using Robinson's estimator and have put forward a pile of very important work.
- In the vein of Robinson's estimator, Edward Kennedy has promoted a series of methods based on Semiparametric Statistics (Influence Functions and Nuisance Tangent Spaces).
- Many other authors are also producing interesting work on Machine Learning in mediation, Regression Discontinuity, Diff-in-Diff. . .
- In general, the asymptotics/technical details are complex in all the work, however recent reviews facilitate some insights into the technical details.

Conclusion.

All the above authors have produced great review articles describing the current state of play.

A recent, perhaps less technical review is[ˆ1] :

Chan, F., & Mátyás, L. (Eds.). (2022). Econometrics with Machine Learning (Vol. 53). Advanced Studies in Theoretical and Applied Econometrics. Springer.

And we will showcase some of this newer work in a forthcoming RADIANCE course.