Machine Learning and Causal Inference

Mixtape Track

MIXTAPE SESSIONS

Where ML fits into causal inference (flashback)

Traditional strategy: $Y_i = \delta D_i + X_i'\beta + \varepsilon_i$, or

1. Regress Y_i on X_i and compute the residuals,

$$\begin{aligned}\tilde{Y}_i &= Y_i - \hat{Y}_i^{OLS}, \\ \hat{Y}_i^{OLS} &= X_i' \left(X'X\right)^{-1} X'Y\end{aligned}$$

2. Regress D_i on X_i and compute the residuals,

$$\begin{aligned}\tilde{D}_i &= D_i - \hat{D}_i^{OLS}, \\ \hat{D}_i^{OLS} &= X_i' \left(X'X\right)^{-1} X'D\end{aligned}$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

Where ML fits into causal inference (flashback)

Traditional strategy: $Y_i = \delta D_i + X_i'\beta + \varepsilon_i$, or

1. Regress Y_i on X_i and compute the residuals,

$$\begin{aligned} \tilde{Y}_i &= Y_i - \hat{Y}_i^{OLS}, \\ \hat{Y}_i^{OLS} &= X_i' \left(X'X\right)^{-1} X'Y \end{aligned}$$

2. Regress D_i on X_i and compute the residuals,

$$\begin{aligned} \tilde{D}_i &= D_i - \hat{D}_i^{OLS}, \\ \hat{D}_i^{OLS} &= X_i' \left(X'X\right)^{-1} X'D \end{aligned}$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

When OLS might not be the right tool for the job:

- there are many variables in X_i
- the relationship between X_i and Y_i or D_i may not be linear

Where ML fits into causal inference (flashback)

ML-augmented regression strategy:

1. Predict Y_i using X_i with ML and compute the residuals,

$$\begin{aligned}\tilde{Y}_i &= Y_i - \hat{Y}_i^{ML}, \\ \hat{Y}_i^{ML} &= \text{prediction generated by ML}\end{aligned}$$

2. Predict D_i using X_i with ML and compute the residuals,

$$\begin{aligned}\tilde{D}_i &= D_i - \hat{D}_i^{ML}, \\ \hat{D}_i^{ML} &= \text{prediction generated by ML}\end{aligned}$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

Where ML fits into causal inference (flashback)

ML-augmented regression strategy:

1. Predict Y_i using X_i with ML and compute the residuals,

$$\begin{aligned}\tilde{Y}_i &= Y_i - \hat{Y}_i^{ML}, \\ \hat{Y}_i^{ML} &= \text{prediction generated by ML}\end{aligned}$$

2. Predict D_i using X_i with ML and compute the residuals,

$$\begin{aligned}\tilde{D}_i &= D_i - \hat{D}_i^{ML}, \\ \hat{D}_i^{ML} &= \text{prediction generated by ML}\end{aligned}$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

Two flavors of machine-assisted causal inference:

1. Post-double selection lasso (PDS lasso), introduced by Belloni, Chernozhukov, and Hansen
2. Double/De-biased machine learning (DML), introduced by Chernozhukov, Chetverikov, Demirer, Duflo, Hansen, Newey, and Robins

Machine-Assisted Causal Inference

- No *identification ex machina*! Still rely on

$$D_i \perp\!\!\!\perp (Y_i(0), Y_i(1)) | X_i$$

- What variables to include in X_i? The omitted variables bias formula is our guide. Uncontrolled (bivariate) regression gives us:

$$\hat{\delta}^{\text{bivariate}} \to \delta + \beta \frac{Cov(D_i, X_i)}{Var(D_i)}$$

  We need to control for variables that

  - affect the outcome
  - are correlated with treatment

- Beware of **bad control**: including post-treatment variables in X_i

PDS Lasso: Preliminaries

Begin with flexible version of our regression model:

$$Y_i = \tau D_i + g\left(X_i\right) + \varepsilon_i$$

Approximate the two CEFs,

$$\begin{aligned} m_D(X_i) &\equiv E[D_i|X_i] \\ m_Y(X_i) &\equiv E\left[Y_i|X_i\right] = \tau m_D(X_i) + g(X_i), \end{aligned}$$

With a **sparse linear approximation**:

$$\begin{aligned} m_Y(X_i) &= X_i'\gamma_Y + r_i \\ m_D(X_i) &= X_i'\gamma_D + s_i, \end{aligned}$$

X_i should contain a **dictionary** of nonlinear transformations like powers and interactions

PDS Lasso: The Recipe

PDS is implemented in three steps:

1. Lasso Y_i on X_i, collect retained features in X_i^Y
2. Lasso D_i on X_i, collect retained features in X_i^D
3. Regress Y_i on D_i and $X_i^Y \cup X_i^D$

Caveats and considerations:

- Standardizing controls pre-lasso is important
- BCH have a formula for the penalty parameter, but cross-validation seems to work just fine
- Inference: just use robust SEs from last step!

Time for python!

DML: Preliminaries

Stick with flexible version of our regression model:

$$Y_i = \tau D_i + g\left(X_i\right) + \varepsilon_i$$

1. Predict Y_i using X_i with ML and compute the residuals,

$$\begin{aligned} \tilde{Y}_i &= Y_i - \hat{Y}_i^{DML}, \\ \hat{Y}_i^{DML} &= \text{prediction generated by ML} \end{aligned}$$

2. Predict D_i using X_i with ML and compute the residuals,

$$\begin{aligned} \tilde{D}_i &= D_i - \hat{D}_i^{DML}, \\ \hat{D}_i^{DML} &= \text{prediction generated by ML} \end{aligned}$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

$\hat{Y}_i^{DML}$ and $\hat{D}_i^{DML}$ should be predictions generated by a machine learning model trained on a set of observations that *does not include i*. We accomplish this via *cross-fitting*

DML: Recipe

1. Divide the sample into K folds
2. For $k = 1, \ldots, K$
   a Train a model to predict Y given X, leaving out observations i in fold k: $\hat{Y}^{-k}(x)$
   b Train a model to predict D given X, leaving out observations i in fold k: $\hat{D}^{-k}(x)$
   c Form residuals $\tilde{Y}_i = Y_i - \hat{Y}^{-k}(X_i)$ and $\tilde{D}_i = D_i - \hat{D}^{-k}(X_i)$
3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

Caveats and considerations:

- Cross-validation to choose tuning parameters
- Inference: use robust SEs from last step

Time for python!

That's a wrap

What I hope you've gotten out of the last couple of days:

- Clarity on distinction between predictive and causal questions
- Foot in the door with python implementations of some common modern supervised machine learning methods
- Tools for using ML methods to control for high dimensional covariates in the service of causal inference

Preview for future workshop:

- Use ML to predict heterogeneous treatment effects (e.g., random causal forests)
- November 10, preview here: github.com/Mixtape-Sessions/Heterogeneous-Effects/

Thank you!