Machine Learning for Economists: Part 5 – [Causal] Inference

Michal Andrle
International Monetary Fund

PRELIMINARY

Washington, D.C.,
November, 2019

Disclaimer #1:

The views expressed herein are those of the authors and should not be attributed to the International Monetary Fund, its Executive Board, or its management.

Context

Machine Learning methods are very popular for **prediction**

Increasingly, there are ways of using machine learning for (causal) statistical **inference**

“Pioneers:”
V. Chernozhukov, A. Belloni, S. Athey, C. Hansen, L.W. Mackey, V. Syrgkanis, S. Wager, G. Imbens, . . .

Two Cultures. . .

[Breiman (2001), Statistical Modeling: The Two Cultures]

Breiman ponders the state of statistics and sees two cultures. . .

- One culture assumes to know the model that supposedly generated the data, tests hypothesis. . .
- The other culture uses algorithmic models and treats the data-generating process as uknown. . .

Breiman argues that committment to the first culture:
“has lead to irrelevant theory, questionable conclusions, and has kept statisticians from working on a large range of interesting current problems”.

“Just Do It” Approach

Machine-Learning/Computer Science community focuses on solving problems. . .

Solve the problem first, worry about all the theory later

Would you rather be “roughly right” or “precisely wrong”?

What use is a linear stylized model with well-understood properties if it fails to solve the problem at hand. . . ?

WORSE:
Classical inference gets often abused by un-disciplined specification searches, and undisciplined data mining. . .
See Leamer (1979): Specification Searches

Machine Learning and [Causal] Inference

Trying not to mix the **statistical** and **causal** issues. . .

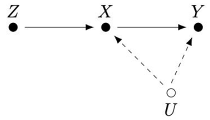
- **Causality** – **why** and **what** is the causal target parameters
  - What is the motivation, what identifies the causal parameters of interest
  - What (not) to condition on (front-door, back-door criteria,. . .)
  - DAGs, Endogeneity, instruments, diff-in-diff, Neyman-Rubin's model, SUTVA, . . .
  - Most oten **untestable** assumptions
- **Statistics** – **how** to estimate the identified target parameters the 'best way'
  - Parametric or non-parametric, functional forms, . . .
  - Model selection, variable selection, especially when $p >> N$
  - Efficient estimation, asymptotic properties, Neyman orthogonality, . . .

Once the 'causal identification' is done, focus on the statistical details. . . (causality $\neq$ inference)

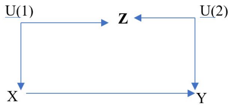
WARNING: Causality – Are you DAGing it?

DAG – Directed Acyclical Graph[s] (J. Pearl (2000))

(A) :)



(B) Should you control for **Z**?



No! Bad covariate, inducing bias due to ‘back-door criterion’.

ML Hopes & Issues

Hopes & Benefits:

- ML methods are very flexible, allow for nonlinearities, . . .
- ML algorithms can deal well with large of variables
- ML is focused on disciplined model selection

Issues & Opportunities:

- **post-selection inference**
- little-to-none formal results for popular ML algorithms (asymptotics, efficiency, . . .)
- rapid progress on statistical inference with ML

For policy analysis both **prediction** and **inference** are important. . .

‘Post-Selection’ Inference. . .

Classical statistical theory ignores model selection in assessment of estimation accuracy. . .

Most statistical theory assumes the model selection is **not adaptive**, not using the data at hand. . . (which makes learning from data really hard!)

Not accounting for the selection process is **bad datamining**, and the inference can be overly optimistic!

‘Replicability crisis’ in science. . .
Ioanidis (2005): “. . . most published research findings are false.”

‘Post-Selection’ Inference and Machine Learning. . .

ML is about GOOD data mining, a disciplined learning from data. . .

Regularization – adaptive model selection

By knowing the process of model selection, it is easier to account for it in the **inference stage**

It matters a lot **what is the subject of inference** – parameters, or more aggreagate ‘effects’ (ATE, etc.)

In some (sub)-models, not all parameters exist!
What do you do?

(conditonal coverage?, simultaneous coverage, controlling for family-wise eror rate?)

Post-Selection Inference using **Sample Splitting**

Barnard (1974) [quoted in Wasserman et al (2018)]:

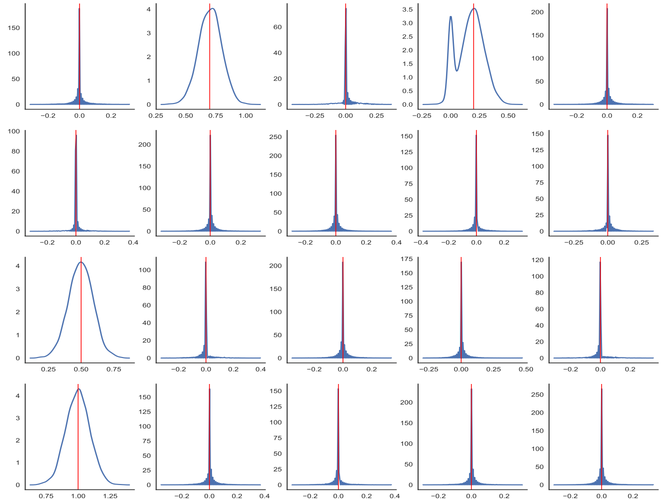
". . . the idea of splitting a sample in two and then developing the hypothesis on the basis of one part and testing it on the remainder may perhaps be said to be one of the most seriously neglected ideas in statistics"

Sample Splitting:

1. Split the IID sample S into S_A and S_B
2. Explore and select models using S_A
3. Given the selected model, carry out inference using S_B

No cheating! . . . dates back to Cox (1975), Stone (1974), etc.)

Model Selection using Sample Splitting



Post-Selection Inference & Bootstrap

Efron (JASA, 2014)

Sometimes*, the whole learning process can be **bootstrapped**!

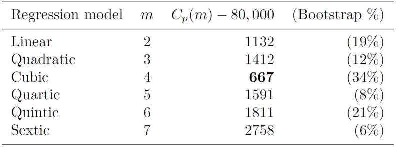
Non-Parametric Bootsrap applied to the whole pipeline of training the model (and parameters) and choosing the model (hyper-parameters, e.g. λ, C_p, lag-length,. . .)

The bootstrap for 'effects' (ATE, means, etc.) is more feasible than for the coefficients (e.g. with sparsity not all are always defined. . .)

* conditions & terms apply ;)

Post-Selection Inference & Bootstrap

Bradley Efron (Estimation and Accuracy after Model Selection, JASA, July 1, 2014)



Regression model	m	$C_p(m) - 80,000$	(Bootstrap %)
Linear	2	1132	(19%)
Quadratic	3	1412	(12%)
Cubic	4	**667**	(34%)
Quartic	5	1591	(8%)
Quintic	6	1811	(21%)
Sextic	7	2758	(6%)

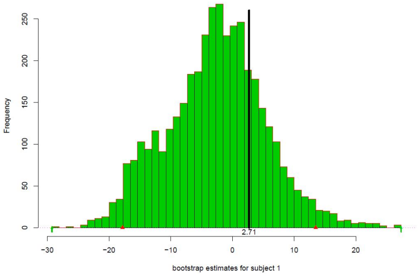


Figure 3: $B = 4000$ bootstrap replications $\hat{\mu}_1^*$ of the C_p-OLS regression estimate for Subject 1. The original estimate $t(\boldsymbol{y}) = \hat{\mu}_1$ is 2.71, exceeding 76% of the replications. Bootstrap standard deviation (2.4) equals 8.02. Triangles indicate 2.5th and 97.5th percentiles of the histogram.

Post-LASSO Inference

An obvious elephant in the room. . .

Lasso is an adaptive model selection algorithm, aimed at prediction accuracy (fit).

Intuitive BUT WRONG!:

1. Train the model using LASSO with CV-alidated λ [**YES**]
2. Estimate post-lasso regressions with selected variables to de-bias the coefficient estimates [**YES**]
3. Carry out standard statistical inference [**NO!**]

What's the way OUT?
Large literature with many special cases. . .

Lee, Sun, Sun, Taylor (AoS, 2016), Fithian, Sun, Taylor (2017), Taylor and Tibshirani (2015), Chaterjee and Lahiri (JASA, 2011), . . .

Post-LASSO Inference

Actually, with LASSO, what is the inference about?

(A) Is the inference about the **coefficients?**

...the trouble is in some models not all coefs exist!

1. Examining everything, $H^*_{0,j} : \beta^*_j = 0$, conditional on **all** variables
2. Inference based on a **sub-model**, $\mathcal{M}$

(B) Or, is the inference about a statistic that is **ALWAYS** part of the model?

- mean prediction, ...
- average treatment effect (ATE), etc.
- ...

Post-LASSO Inference

A few suggestions. . .

- **Sample splitting. . .**
- **'In Defense of the Indefensible'**

  (Zhao, Shojaie, Witten, 2017) The naive two-step approach shouldn't work... but 'can' yield confidence intervals with asympt. correct coverage, as well as OK p-vals; and there's reason for that. . .
- **Various form of bootstrap**

  Chaterjee and Lahiri (2011) (resid. bstrap), Efron (2014)
- **Exact Post-selection inference** conditioned on the selection event Lee, Sun, Sun, Taylor (2016)

  When the variable is easily selected, the intervals are essentially the OLS intervals, but when a variable is barely selected, things are bad and intervals very wide. . .
- **Go Bayesian!**

High-Dimensional Inference & Treatment Effects…

Different focus here – **ONE** parameter of interest, the rest are *nuisance* params…

Non-Technical Introduction

Parametric Variable Selection & Post-Selection Inference

Given K variables, $\{X_1, X_2, \ldots, X_k\}$, select 'optimally' only R components for a set Ω

$$y_i = \alpha_1 \times \text{Treatment}_i + \sum_{r \in \Omega} \beta_r X_r \qquad (1)$$

Semi-Parametric Estimation & Variable Selection

$$y_i = \theta \times \text{Treatment}_i + f(\mathbf{X}_i), \qquad (2)$$

with $f(.)$ uknown and **X** high-dimensional. . .

Non-Parametric

Orthogonal/Double Machine Learning

Chernozhukov et al. (2018)

Consider a problem

$$Y = \theta \times D + f(X) + U, \tag{3}$$

with $E[U|X, D] = 0$.

Y	–	outcome variable of interest
D	–	policy or treatment variable
θ	–	target parameter of interest
X	–	high-dimensional covariates (confounders)
$f(.)$	–	unkown, complicated function

Confounders X important because

$$D = m(X) + V, \qquad E[V|X] = 0. \tag{4}$$

How to use modern ML techniques to estimate $f(.)$ and carry out inference about low-dimensional θ?

Orthogonal/Double Machine Learning

Frisch-Waugh-Lovell style...

Rewrite the model in expectations, conditioned on X, i.e.

$$E[Y|X] = \theta \times E[D|X] + f(X) + E[U|X], \tag{5}$$

and subtract from the origional problem

$$Y = \theta \times D + f(X) + U, \tag{6}$$

to get

$$\underbrace{Y - E[Y|X]}_{R_Y} = \theta \underbrace{(D - E[D|X])}_{R_D} + \underbrace{(U - E[U|X])}_{\widehat{U}.}$$

The estimate of θ is obtained from regressing R_Y on R_D.

Orthogonal/Double Machine Learning

The regression of R_Y on R_D is *infeasible*…

Use modern **machine-learning tools** (random forests, neural nets, boosting, Lasso, …) to learn mappings functions $h(.)$ and $m(.)$

- $\widehat{R}_Y \equiv (Y - E[Y|X]) \equiv Y - \widehat{h}(X)$
- $\widehat{R}_D \equiv (Y - E[D|X]) \equiv Y - \widehat{m}(X)$

Can we estimate θ as

$$\hat{\theta} = (\widehat{D}'_D \widehat{R}_D)^{-1} \widehat{R}'_D \widehat{R}_Y \text{ ?} \tag{7}$$

The issue is that $\widehat{h}(.)$ and $\widehat{m}(.)$ were obtained using the whole sample, risking over-fitting and complicating inference…

Sample Splitting & Cross-Fitting

Sample splitting is one of the most **understated** methods in statistics. . .

1. Split the sample in two parts, S_1 and S_2. . .
2. Use S_1 to train and select the models
3. Use S_2 to carry out inference using the models

Sample splitting decreases sample size and thus **lowers power**.

Cross-fitting: Do multiple splits and average the estimates. . .

Of course, related to cross-validation. . .

Orthogonal/Double Machine Learning + Cross-Fitting

Chernozhukov et al. (2018)

Given the model

$$Y = \theta \times D + f(X) + U, \tag{8}$$

1. Randomly split the sample in two parts, S_1 and S_2.
2. Using the sample S_1, estimate $E[Y|X] = \widehat{h}(X)$ and $E[D|X] = \widehat{m}(X)$.
3. Using the sample S_2 compute projection errors $\widehat{R}_Y = Y - \widehat{h}(X)$ and $\widehat{R}_D = D - \widehat{m}(X)$.
4. Compute the estimate $\widehat{\theta}_1$ by regressing $\widehat{R}_Y$ on $\widehat{R}_D$.
5. Flip the roles of samples S_1 and S_2, to estimate $\widehat{\theta}_2$.
6. Estimate $\widehat{\theta} = \frac{1}{2}\widehat{\theta}_1 + \frac{1}{2}\widehat{\theta}_2$. Now, $\sqrt{N}(\widehat{\theta} - \theta) \sim N(0, \Sigma)$

Orthogonal ML – Monte Carlo Simulation

Inspired by the blog by Gabriel Vasconselos…

Data-Generating Process to test:

$$
\begin{aligned}
y_i &= D_i\theta + \cos^2(x_i'\gamma) + u_i && (9)\\
D_i &= \sin(x_i'\gamma) + \cos(x_i'\gamma) + v_i && (10)
\end{aligned}
$$

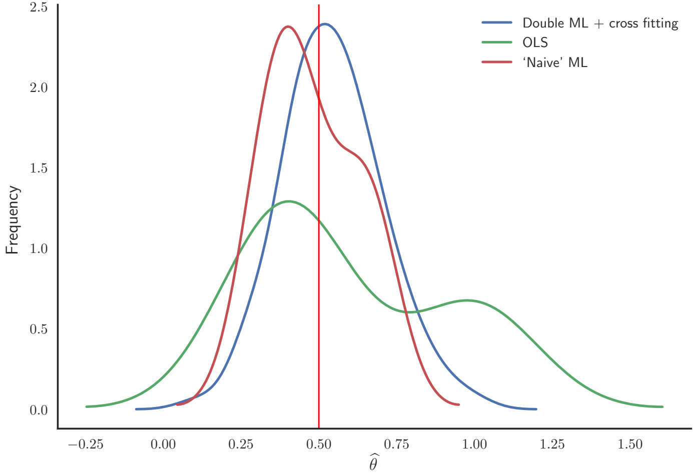
with $u, v \sim N(0,1)$, $\theta = 0.5$ and $\gamma_k = 1/k$.

Comparing three estimators:

- OLS
- 'Naive' ML, see Chernozhukov et al (2016) for details
- Double-ML with cross-fitting, using Random Forests for regression

Note: for the OLS estimation the "safer" benchmark is $\texttt{y} \sim \texttt{d} + \texttt{x}$

Orthogonal ML – Monte Carlo Simulation



Orthogonal ML – Examples

Orthogonal ML – Why Does It Work?

Magic. . . ;)

A few of key ingredients (intuitively):

- The implied moment conditions (and the influence function) are not “too” sensitive to “small” mistakes in estimating nuisance functions $h(.)$ and $m(.)$
- Not every moment condition necessarily satisfies this ‘**Neyman Orthogonality**’
- Sample splitting further helps to lower assumptions needed about smoothness of $h(.)$ and $m(.)$
- The FWL “residualization” is essentially the same as in standard semi-parametric literature. BUT the inference works for quite different reasons with ML plugin estimators, which have more complex properties. . .

Double ML – General version

Double-Selection Approach to Lasso (1)

Belloni, Chernozhukov, and Hansen (2014)

Consider having sample size N for

$$Y_i = \theta \times D_i + X_i'\alpha + \varepsilon_i \tag{11}$$

where

Y_i – outcome
θ – [causal] coefficient of interest
D_i – treatment, exogenous variable
X_i – “large” vector of covariates (confounders), $P >> N$
α – ‘nuisance’ parameters

With $P >> N$, we must work on **variable selection**, standard OLS won’t do it...

What pops into your mind? **LASSO!**

Double-Selection Approach to Lasso (2)

Bad Approach:
Running standard Lasso, while forcing D to be always selected, will work poorely.

- Lasso does not necessarilly always selects all the ‘correct’ variables
- Lasso targets prediction and will omit variables that are highly correlated with D, as they are not needed for prediction…
- This may result into severe *omitted variables bias*.
- In general, bootstrap won’t help…

GOOD Approach: Double-Selection Lasso

Double-Selection Approach to Lasso (3)

GOOD Approach: Double-Selection Lasso

1. Run first-stage selection using Lasso:

$$Y_i = X_i'\beta_1 + \nu_{1,i} \tag{12}$$

2. Run second-stage selection using Lasso:

$$D_i = X_i'\beta_2 + \nu_{2,i} \tag{13}$$

3. Run OLS with the **union of variables selected** in Stage 1 and Stage 2: $\tilde{X}_i \in S|(\beta_{1,i} \neq 0 \text{ or } \beta_{2,i} \neq 0)$:

$$Y_i = \alpha D_i + \tilde{X}_i'\theta + \varepsilon \tag{14}$$

Double-Selection Lasso: Simulation Experiment

[Follows Chernozhukov et al. 2016]

$$
\begin{aligned}
Y_i &= D_i\theta + X_i'\alpha_1 + u_i \qquad (15)\\
D_i &= X_i'\alpha_2 + v_i \qquad (16)
\end{aligned}
$$

X is $[N_{obs} \times k]$

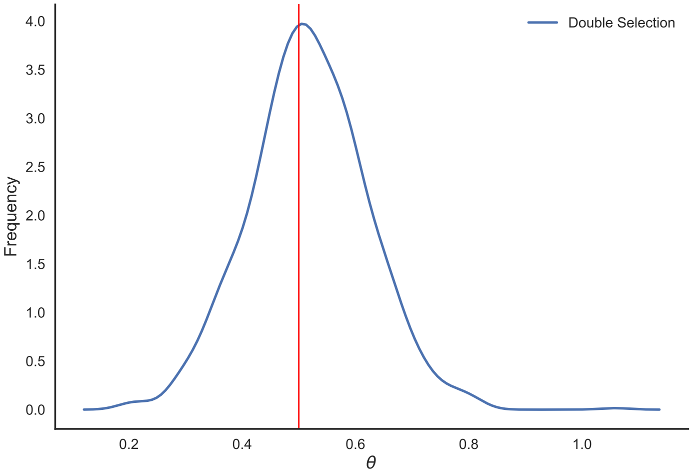
The model is “aproximately sparse”, i.e.

$$\alpha_{1,j} = 1/j^2, \alpha_2 = 0.7\alpha_1 \;\; j = 1, \ldots, k$$

The covariates are correlated and Normally distributed

$$x \sim N(0, \Sigma), \Sigma_{kj} = (1/2)^{|j-k|}$$

Double-Selection Lasso: Simulation Experiment ($p >> N$)



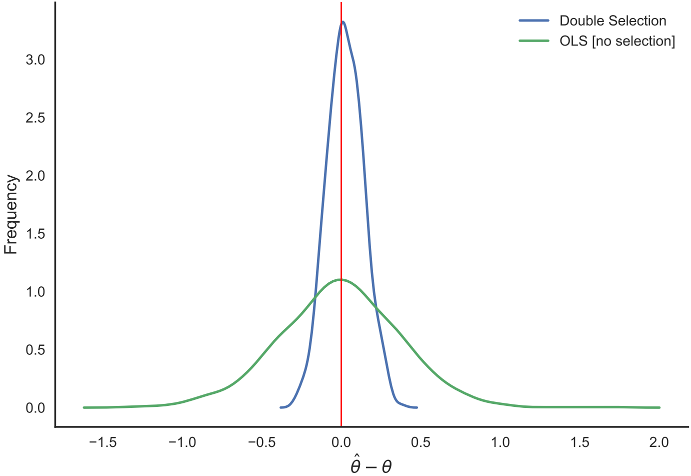
Double-Selection Lasso: Simulation Experiment ($p < N$)

When $p < N$ the OLS is still feasible.

Of course, as degrees of freedom decline, the variance of the estimator of θ increases dramatically…

Variable/model selection using *post double-selection* is vastly superior…

Double-Selection Lasso: Simulation Experiment ($p < N$)



]

Double-Selection Lasso: Instrumental Variables (IV)

Chernozhukov, Hansen, and Spindler (2015)

Consider the following linear IV model:

$$
\begin{align}
Y_i &= \alpha D_i + X_i'\beta + \varepsilon_i, \tag{17} \\
D_i &= X_i'\gamma + Z_i'\delta + u_i, \tag{18} \\
Z_i &= \Pi X_i + \zeta_i \tag{19}
\end{align}
$$

where

Y_i	–	outcome for *i*-th observation
D_i	–	**endogenous** variable of interest
X_i	–	potentially high-dim. vector of covariantes
Z_i	–	potentially high-dim. vector of **instruments**
α	–	parameter of interest

Assume the dimension of full parameter vector is much larger than estimate with available sample size. . .

Double-Selection Lasso: Instrumental Variables (IV)

Using simple substitutions, the system

$$\begin{aligned} Y_i &= \alpha D_i + X_i'\beta + \varepsilon_i, & (20)\\ D_i &= X_i'\gamma + Z_i'\delta + u_i, & (21)\\ Z_i &= \Pi X_i + \zeta_i & (22) \end{aligned}$$

can be expressed as a reduced form

$$\begin{aligned} Y_i &= X_i'\theta + r_i^Y, & (23)\\ D_i &= X_i'\psi + r_i^D, & (24) \end{aligned}$$

which depends only on X_i.

Post-lasso double selection strategy that ‘immunizes’ estimation from selection errors exists. . .

Double-Selection Lasso: Instrumental Variables (IV)

Algorithm:

1. Do post-lasso regression of D_i on X_i, Z_i to get $\hat{\gamma}$ and $\hat{\delta}$.
2. Do post-lasso regression of Y_i on X_i to get $\hat{\theta}$
3. Define $\widehat{D}_i := X_i'\hat{\gamma} + Z_i'\hat{\delta}$ and run post-lasso of $\widehat{D}_i$ on X_i, getting $\hat{\psi}$
4. Define

$$\begin{aligned} \hat{r}_i^Y &:= Y_i - X_i'\hat{\theta}, & (25) \\ \hat{r}_i^D &:= D_i - X_i'\hat{\psi}, & (26) \\ \hat{\upsilon}_i^D &:= (X_i'\hat{\gamma} + Z_i'\hat{\delta}) - X_i'\hat{\psi}. & (27) \end{aligned}$$

5. Estimate $\hat{\alpha}$ by **standard** IV regression of $\hat{r}_i^Y$ on $\hat{r}_i^D$, using $\hat{\upsilon}_i$ as an instrument. Standard inference applies...

Double-Selection Lasso: Instrumental Variables (IV)

Why this works?

It works because the implied moment condition for the last-stage IV problem

$$\mathbf{E}[(\hat{r}_i^Y - \hat{r}_i^D \alpha)\hat{\upsilon}] = 0 \tag{28}$$

is not overly sensitive (‘immunized‘) against lasso selection mistakes, and imperfect estimation of $\gamma, \delta, \psi, \beta$.

Definition of $\hat{\upsilon}_i^D := (X_i'\hat{\gamma} + Z_i'\hat{\delta}) - X_i'\hat{\psi}$ is very important.

Chernozhukov, Hansen, and Spindler (2015) illustrate that moment condition

$$\mathbf{E}[(\hat{r}_i^Y - \hat{r}_i^D \alpha)\widehat{D}_i] \equiv \mathbf{E}[(\hat{r}_i^Y - \hat{r}_i^D \alpha)(X_i'\hat{\gamma} + Z_i'\hat{\delta})] = 0 \tag{29}$$

is not ‘robust’ to selection errors…

Double-Selection Lasso: Simulation Experiment

Digression: Neyman-Rubin Potential-Outcomes Model

Let's consider a **binary treatment** $D \in \{0, 1\}$.

Fundamental Problem of Causal Inference:

We never observe the effects of treatment **and** non-treatment for the given individual (or unit)...

Potential Outcomes:

- $y_i(0)$ – outcome if unit i **not treated**, $D = 0$
- $y_i(1)$ – outcome if unit i **treated**, $D = 1$

We observe either $y_i(0)$ or $y_i(1)$ but never both. We observe one outcome, the other being **counter-factual**.

Digression: Neyman-Rubin Potential-Outcomes Model

Treatment effect: $\tau = y_i(1) - y_i(0)$

It is impossible to learn the treatment effect using the observed data without additional assumptions...

The assumptions differ for:

(a) **Randomized controlled trials (RCT)**
(treatment D is random, independent of outcome and control variables...)

(b) **Observational studies**
(treatment D may depend on control variables, X...)

Digression: Neyman-Rubin Potential-Outcomes Model

Unconfoundedness Assumption
Treatment assignment uncondounded when treatment is independent of potential oucomes, **after** conditioning on controls (confounders)

Intuitively, one searches for treated unit *j* that's "as similar as possible" to a non-treated unit *i* and compare their outcomes...

Conditional Average Treatment Effect (CATE)

$$\begin{aligned}\text{CATE}(x) = \tau(x) &= E[Y|D=1, X=x] - E[Y|D=0, X=x] \\ &= E[Y(1)|D=1, X=x] - E[Y(0)|D=0, X=x] \\ &= E[Y(1)|X=x] - E[Y(0)|X=x] \qquad (30)\end{aligned}$$

Average Treatment Effect (ATE)

$$\text{ATE} = E[CATE(x)] \qquad (31)$$

ML usage...

Let $p = \text{pr}(D_i = 1)$ be the marginal treatment probability, and let

$$e(x) = \text{pr}(D_i = 1 | X_i = x) \tag{32}$$

be the **propensity score** (conditional treatment prob.)

One option for using ML is to estimate flexible predictive models for the propensity score. . . [low-hanging fruit] for propensity-score matching

Causal Forests

Stefan Wager and Susan Athey (2018)

Using random forest to estimate heterogeneous treatment effects...

$$\tau(x) = E[y_i(1) - y_i(0)|X_i = x]. \tag{33}$$

The point is to allocate “similar as possible” units, treated and non-treated, to the trees’ leafs and read-off the treatment effect within the leaf...

Trees are very good at creating sets of ‘similar’ units, with the similarity being defined with respect to outcome or probability of treatment...

Causal trees make sure each leaf has at least k observations from both treated and non-treated groups

Causal Forests

Stefan Wager and Susan Athey (2018)

Having observed data (y_i, x_i, D_i), we want to estimate

$$\tau(x) = E[y_i(1) - y_i(0)|X_i = x]. \tag{34}$$

Given a **tree** with leaves $L_k,\ k = 1, \ldots, K$ we estimate CATE(x) as difference of average outcome of treated and non-treated units **within the leaf** where the unit with control features x is a member of.

$$\widehat{\tau}(x) = \text{avg}(y_i)_{\{i:D_i=1, x_i \in L\}} - \text{avg}(y_i)_{\{i:D_i=0, x_i \in L\}} \tag{35}$$

Causal Forests

Random forests can be viewed as ‘adaptive nearest neighbors’

What is the difference to k-NN then?

Trees and forests do not measure ‘closeness’ just based on X, but mainly by their effects on outcome (prob. of treatment, say. . .). Closeness is defined as being member of the same leaf of a decision tree. . .

This allows the implicit functional form defining closeness to reflecting the strength of the information (signal)

TODO: Generalized Random Forests and ”weigted adaptive NN”

Causal Forests

Honest Trees
In order to get valid inference, Wager and Athey (2018) require trees to be ‘honest’

- Honesty is related to over-fitting.
- Honesty means that the treatment effect is computed using data y_i that were NOT used for training the model…
- Honesty can be achived by **sample splitting** (sample splitting again!)

Causal Forests

Stefan Wager and Susan Athey (2018)

Athey and Wager propose two types of **causal trees**

- **Double-Sample Trees**
  1. Split sample in halves, S_A and S_B.
  2. Use sample S_A to train the splits for trees
  3. Use sample S_B for within-leaf estimation, given the tree structure
- **Propensity Trees** (Also Wang et al. 2015)
  1. Train a **classification tree**, with X_i predicting treatment, D_i
  2. Each leaf must have at least k observations on treated and non-treated units, each
  3. Estimate $\tau(x)$ within the leaf containing x

Propensity trees training uses no information on outcome, Y_i.

Causal Forests

Stefan Wager and Susan Athey (2018)

Causal forests use a splitting criterion that maximizes the **variance** of treatment effects, $\widehat{\tau}(x)$.

Wager and Athey (2018) motivate this by an analogy of regression trees, where splits are done to minimize the mean-square error…

Finding the split that minimizes the mean-square error amounts to maximizing the variance of the prediction…

$$\sum_{i \in L} (\hat{\mu}(x_i) - Y_i)^2 = \sum_{i \in L} (Y_i^{obs})^2 - \sum_{i \in L} \hat{\mu}(x_i)^2$$

Causal Forests and Trees

Implementation:

Implementation in R, Python, or Matlab straightforward. . .

Ahrens et al.(2018) implement rich lasso toolbox in **Stata**:

- post double-selection lasso (pdslasso)
- elastic net, ridge, lasso, cross-validation, . . .
- . . .

References

- Belloni, Chernozhukov, Hansen (2014): High-Dimensional Methods and Inference on Structural and Treatment Effects, JoEconomic Perspectives, Vol. 28, No. 2, Spring 2014, pp. 29–50
- Ahrens, A., C.B. Hansen, M.E. Schaffer (2018): LASSOPACK and PDSLASSO: Prediction, model selection, and causal inference with regularized regression
- Instrumental Variables Estimation with Very Many Instruments and Controls, 2015
- Chernozhukov, V., D. Chetverikov, E. Duflo, Ch. Hansen, M. Demirer, W. Newey, and J. Robins (2018): Double/debiased machine learning for treatment and structural parameters, The Econometrics Journal, vol 21(1), pp. C1-C68
- Wager, S. and S. Athey (2018):Estimation and Inference of Heterogeneous Treatment Effects using Random Forest, Journal of the American Stat. Association, vol. 113, n. 523, 1228-1242

Thank you for your patience. . .