Causal Machine Learning in Economics

A. Colin Cameron
U.C.-Davis

.

Presented at Big Ag Data Conference
University of California - Davis

January 10 2020

Causal Machine Learning in Economics

- Current **causal** inference methods in microeconometric applications have a **high-dimensional component**
  - e.g. estimate a key parameter assuming selection on observables only
    - good controls makes this assumption more reasonable
  - e.g. IV with many available instruments
    - good few instruments avoids many instruments problem
  - e.g. a structural dynamic discrete choice model with many potential states.
- Standard nonparametric and semiparametric methods suffer from a course of dimensionality.
- **Machine learning methods** do better in many applications
  - though valid statistical inference needs to control for this data mining.

Outline

1. Partial Linear Model

- A **partial linear** control function model specifies

$$y = \mathbf{d}'\boldsymbol{\alpha} + g(\mathbf{x}_c) + u \text{ where } g(\cdot) \text{ is unknown.}$$

- Here
  - $\mathbf{d}$ are policy or treatment variables of interest
    - for simplicity we will later focus on the scalar case
  - $\mathbf{x}_c$ are control variables
  - $g(\cdot)$ is an unknown function
- Selection on observables assumption
  - consistent OLS estimation of $\boldsymbol{\alpha}$ requires $E[u|\mathbf{d}, \mathbf{x}_c] = 0$
  - this is more plausible the better is $g(\mathbf{x}_c)$.

Curse of dimensionality kills standard semiparametric methods

- Robinson (1988) proposed **semiparametric estimation**

$$y = \mathbf{d}'\alpha + g(\mathbf{x}_c) + u \text{ where } g(\cdot) \text{ is unknown.}$$

  - Kernel regression of y on $\mathbf{x}_c$ gives residual $u_{y|\mathbf{x}_c}$
  - Kernel regression of $\mathbf{d}$ on $\mathbf{x}_c$ gives residuals $u_{\mathbf{d}|\mathbf{x}_c}$
  - OLS of $u_{y|\mathbf{x}_c}$ on $u_{\mathbf{d}|\mathbf{x}_c}$ gives root-N consistent asymptotically normal $\widehat{\alpha}$.
- This works if $\mathbf{x}_c$ is of low dimension
  - e.g. y = energy consumption; $\mathbf{d}$ = usual demand determinants; $\mathbf{x}_c$ is time of day (scalar).
- Instead consider $\mathbf{x}_c$ is of high dimension - many controls
  - kernel regression fails due to **curse of dimensionality**.
- Solution: use a machine learner rather than kernel regression
  - here use the LASSO instead of kernel regression.

LASSO (Least Absolute Shrinkage And Selection)

- The basic **LASSO estimator** $\widehat{\boldsymbol{\beta}}_\lambda$ of $\boldsymbol{\beta}$ minimizes

$$Q_\lambda(\boldsymbol{\beta}) = \sum_{i=1}^{n}(y_i - \mathbf{x}_i'\boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

  - where $\lambda \geq 0$ is a tuning parameter to be determined
  - and $\mathbf{x}'s$ are standardized to have mean 0 and variance 1.
- The idea is to penalize model complexity
  - this induces bias but can reduce variance.
- LASSO sets many $\boldsymbol{\beta}$ to zero and shrinks remaining towards zero
  - hence name.
- Tuning parameter λ is most often determined by MSE cross-validation or AIC or BIC
  - but in this causal partial linear application we want a tighter penalty
    - ★ like oversmoothing with kernels
  - Chernozhukov et al. propose a particular data-dependent value of λ.

LASSO (left) versus Ridge

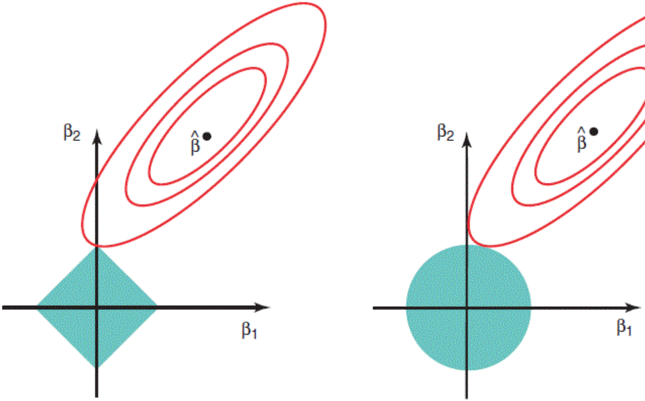

FIGURE 6.7. *Contours of the error and constraint functions for the lasso* (left) *and ridge regression* (right). *The solid blue areas are the constraint regions,* $|\beta_1| + |\beta_2| \leq s$ *and* $\beta_1^2 + \beta_2^2 \leq s$, *while the red ellipses are the contours of the RSS.*

Partialling out LASSO for Partial Linear Model

- Let $g(\mathbf{x}_c) = \mathbf{x}'\gamma$ where $\mathbf{x}$ are flexible transformations of $\mathbf{x}_c$ such as polynomials and interactions.
- Then
$$y = \alpha d + \mathbf{x}'\gamma + u \text{ where } g(\cdot) \text{ is unknown.}$$
- Partialling out LASSO estimator (scalar d)
  - Lasso regression of y on $\mathbf{x}$ gives residual $u_{y|\mathbf{x}}$
  - Lasso regression of d on $\mathbf{x}$ gives residual $u_{\mathbf{d}|\mathbf{x}}$
  - OLS of $u_{y|\mathbf{x}}$ on $u_{\mathbf{d}|\mathbf{x}}$ gives root-N consistent asymptotically normal $\widehat{\boldsymbol{\alpha}}$.
- Implementation
  - requires only LASSO and OLS
  - most machine learning is in R
  - Stata 16 introduced LASSO, Ridge, elasticnet and extensions
  - Also there is a Stata addon `pdslasso` for this problem.

Example

- Example with $N = 2955$, d is scalar, $\dim(\mathbf{x}) = 176$.
  - $y =$ `ltotexp` is log annual medical expenditure for people aged 65-90
  - $d =$ `suppins` is indicator for supplementary health insurance (beyond basic Medicare)
  - $\mathbf{x} =$ 176 variables created from levels, quadratics and interactions of 5 continuous and 13 binary variables.

```
. use mus203mepsmedexp.dta, clear

. keep if ltotexp != .
(109 observations deleted)

. global xlist2 income educyr age famsze totchr

. global dlist2 female white hisp marry northe mwest south ///
>     msa phylim actlim injury priolist hvgg

. global rlist2 c.($xlist2)##c.($xlist2) i.($dlist2) c.($xlist2)#i.($dlist2)
```

Example estimated in Stata 16

```
. * Partialling out partial linear model using default plugin lambda

.
. poregress ltotexp suppins, controls($rlist2)

Estimating lasso for ltotexp using plugin
Estimating lasso for suppins using plugin

Partialing-out linear model          Number of obs               =      2,955
                                     Number of controls          =        176
                                     Number of selected controls =         21
                                     Wald chi2(1)                =      15.43
                                     Prob > chi2                 =     0.0001

------------------------------------------------------------------------------
             |               Robust
     ltotexp |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     suppins |   .1839193   .0468223     3.93   0.000     .0921493    .2756892
------------------------------------------------------------------------------

Note: Chi-squared test is a Wald test of the coefficients of the variables
      of interest jointly equal to zero. Lassos select controls for model
      estimation. Type lassoinfo to see number of selected variables in each
      lasso.
```

2. Orthogonalization

- The preceding method works because estimation is based on an orthogonalized moment.
- Define $\boldsymbol{\alpha}$ as parameters of interest and $\boldsymbol{\eta}$ as nuisance parameters.
- Estimate $\widehat{\boldsymbol{\alpha}}$ is obtained following first step estimate $\widehat{\boldsymbol{\eta}}$ of $\boldsymbol{\eta}$

$$\begin{array}{ll} \text{First stage:} & \widehat{\boldsymbol{\eta}} \text{ solves} \sum_{i=1}^{n} \omega(\mathbf{w}_i, \boldsymbol{\eta}) = \mathbf{0} \\ \text{Second stage:} & \widehat{\boldsymbol{\alpha}} \text{ solves} \sum_{i=1}^{n} \psi(\mathbf{w}_i, \boldsymbol{\alpha}, \widehat{\boldsymbol{\eta}}) = \mathbf{0}. \end{array}$$

- Noise in estimating $\boldsymbol{\eta}$ usually effects the asymptotic distribution of $\widehat{\boldsymbol{\alpha}}$
  - e.g. Heckman two-step estimator in selection models.
- But this is not always the case
  - e.g. feasible GLS asymptotic distribution not affected by first-stage estimation to get $\widehat{\Omega}$.

Orthogonalization (continued)

- Result: first-stage estimation of $\boldsymbol{\eta}$ does not effect the second-stage asymptotic distribution of $\widehat{\boldsymbol{\alpha}}$ if the second-stage function $\psi(\cdot)$ satisfies

$$E[\partial\psi(\mathbf{w}_i, \boldsymbol{\alpha}, \boldsymbol{\eta})/\partial\boldsymbol{\eta}] = \mathbf{0}$$

- Intuition: on average changing $\boldsymbol{\eta}$ does not change $\psi(\cdot)$.
- Proof: see next slide.

Orthogonalization (continued)

- Why does this work?

$$\begin{aligned} & \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \psi(\mathbf{w}_i, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\eta}}) \\ = \;& \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \psi(\mathbf{w}_i, \boldsymbol{\alpha}_0, \boldsymbol{\eta}_0) + \frac{1}{n} \sum_{i=1}^{n} \left. \frac{\partial \psi(\mathbf{w}_i, \boldsymbol{\alpha}, \boldsymbol{\eta})}{\partial \boldsymbol{\alpha}'} \right|_{\boldsymbol{\alpha}_0, \boldsymbol{\eta}_0} \times \sqrt{n}(\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0) \\ & \quad + \frac{1}{n} \sum_{i=1}^{n} \left. \frac{\partial \psi(\mathbf{w}_i, \boldsymbol{\alpha}, \boldsymbol{\eta})}{\partial \boldsymbol{\eta}'} \right|_{\boldsymbol{\alpha}_0, \boldsymbol{\eta}_0} \times \sqrt{n}(\widehat{\boldsymbol{\eta}} - \boldsymbol{\eta}_0) \end{aligned}$$

- By a law of large numbers $\frac{1}{n} \sum_{i=1}^{n} \left. \frac{\partial \psi(\mathbf{w}_i, \boldsymbol{\alpha}, \boldsymbol{\eta})}{\partial \boldsymbol{\eta}} \right|_{\boldsymbol{\alpha}_0, \boldsymbol{\eta}_0}$ converges to its expected value which is zero if $E[\partial \psi(\mathbf{w}_i, \boldsymbol{\alpha}, \boldsymbol{\eta}) / \partial \boldsymbol{\eta}] = \mathbf{0}$.
- So the term involving $\widehat{\boldsymbol{\eta}}$ drops out.
- For more detail see Cameron and Trivedi (2005, p.201).

Orthogonalization in partial linear model

- Consider the partially linear model and manipulate

$$\begin{array}{rrcll} & y & = & \alpha d + \mathbf{x}'\gamma + u & \text{where } E[u|x_1, \mathbf{x}] = 0 \\ \Rightarrow & E[y|\mathbf{x}] & = & \alpha E[d|\mathbf{x}] + \mathbf{x}'\gamma & \text{as } E[u|\mathbf{x}] = 0 \\ & y - E[y|\mathbf{x}] & = & \alpha(d - E[d|\mathbf{x}]) + u & \text{subtracting} \end{array}$$

- Recall that OLS of y on $\mathbf{x}$ has f.o.c. $\sum_i x_i u_i = 0$
  - so is sample analog of population moment condition $E[xu] = E[x(y - \beta x)] = 0$.
- Partialling out estimator therefore solves population moment condition

$$E[(d - E[d|\mathbf{x}])\{(y - E[y|\mathbf{x}]) - \alpha(d - E[d|\mathbf{x}])\}] = 0.$$

- Then $E[\psi(\cdot)] = 0$ where define

$$\begin{array}{rcl} \psi(\cdot) & = & (d - \eta_1)\{(y - \eta_2) - \alpha(d - \eta_1)\} \\ \text{where } \eta_1 & = & E[y|\mathbf{x}] \text{ and } \eta_2 = E[y|\mathbf{x}] \end{array}$$

Orthogonalization in partial linear model (continued)

- Estimation is based on $E[\psi(\mathbf{w}, \alpha, \eta_1, \eta_2)] = 0$ where

$$\begin{aligned} \psi(\cdot) &= (d-\eta_1)\{(y-\eta_2)-\alpha(d-\eta_1)\} \\ \text{where } \eta_1 &= E[y|\mathbf{x}] \text{ and } \eta_2 = E[y|\mathbf{x}] \end{aligned}$$

- This satisfies the orthogonalization condition since
  - $E[\partial\psi(\mathbf{w}, \alpha, \boldsymbol{\eta})/\partial\eta_1] = E[2\alpha(d-\eta_1)-(y-\eta_2)] = 0$
    - as $\eta_1 = E[d|\mathbf{x}]$ and $\eta_2 = E[y|\mathbf{x}]$
  - $E[\partial\psi(\mathbf{w}, \alpha, \boldsymbol{\eta})/\partial\eta_2] = E[-(d-\eta_1)] = 0$
    - as $\eta_1 = E[y|\mathbf{x}]$.

3. Cross Fitting

- The partialling-out LASSO method requires a sparsity assumption that the number of nonzero coefficients of **x** grows at rate no more than $\sqrt{N}$
  - more precisely $s/(\sqrt{N}/\ln p)$ should be small where
    - $s = \#$variables in true model
    - $p = \#$potential regressor.
- This rate of convergence improves to N if sample splitting is used
  - estimate nuisance parameters $\boldsymbol{\eta}$ using part of the sample (e.g. 90%)
  - estimate α using the remaining part of the sample (e.g. 10%).
- A variation uses the entire sample to estimate α as explained next.

K-fold cross fitting (continued)

- Consider the case $K = 10$
- With 10 folds or partitions of the data do for $k = 1, ..., 10$
  - estimate nuisance parameters η (here LASSO) using all but fold k
  - use these to form residuals in just fold k
- Combine these 10 sets of residuals so have residuals for all observations
  - OLS regress $u_{y|\mathbf{x}}$ on $u_{d|\mathbf{x}}$.
- This is Stata 16 command `xporegress`
- Using orthogonalization and cross-fitting is called
  - Double machine learning
  - Debiased machine learning
  - Neyman machine learning (after Neyman's 1959 c-alpha test).

4. Further Discussion

- In principle the preceding approach of orthogonalization and cross-fitting works for any machine learner, not just LASSO
  - ridge regression, neural networks and random forests
  - though assumptions and convergence rates may vary.
- The preceding example can be adapted to allow $\mathbf{d}$ to be endogenous
  - in Stata `poivregress`
- In principle the preceding approach applies to any orthogonalization condition in a "regular" model
  - though there may be an efficiency loss in using an orthogonalization condition.
- In particular in the binary treatment model with heterogeneous effects a standard estimator of ATE and ATT is the doubly-robust estimator of Robins and Roznitsky (1995) and Hahn (1998)
  - this satisfies the orthogonalization condition.

5. A Very Few References

- My webpage has slides of several talks plus references
  - http://cameron.econ.ucdavis.edu/e240f/machinelearning.html
- Undergraduate / Masters level book
  - **ISL:** Gareth James, Daniela Witten, Trevor Hastie and Robert Tibsharani (2013), *An Introduction to Statistical Learning: with Applications in R, Springer*.
  - free legal pdf at http://www-bcf.usc.edu/~gareth/ISL/
  - $25 hardcopy via www.springer.com/gp/products/books/mycopy
- Accessible paper on LASSO for partial linear and many instrument IV
  - Alex Belloni, Victor Chernozhukov and Christian Hansen (2014), "High-dimensional methods and inference on structural and treatment effects," *Journal of Economic Perspectives*, Spring, 29-50.
- Key paper on double machine learning
  - Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey and James Robins (2018), "Double/debiased machine learning for treatment and structural parameters," *The Econometrics Journal*, 21, C1-C68.

- Susan Athey has a lot. She emphasizes random forests and heterogeneous effects.
  - Stefan Wager and Susan Athey (2018), "Estimation and Inference of Heterogeneous Treatment Effects using Random Forests," JASA, 1228-1242.
- A paper with great detail on the current literature with many references.
  - Susan Athey and Guido Imbens (2019), "Machine Learning Methods Economists Should Know About."
- Applied economics focusing on prediction
  - Sendhil Mullainathan and J. Spiess: "Machine Learning: An Applied Econometric Approach", *Journal of Economic Perspectives*, Spring 2017, 87-106.
- Forthcoming book chapter
  - Colin Cameron and Pravin Trivedi (2020), "Machine Learning for Prediction and Inference", chapter 28 in *Microeconometrics using Stata*, second edition, forthcoming.