Econometric Foundations of ‘riplDML’, A Double Machine Learning Library

Akiva Yonah Meiselman

RIPL

November 9, 2022

Overview

Double/Debiased Machine Learning (DML)

- A method of controlling for a high-dimensional nuisance parameter; a variable selection method
- Is it double or debiased or both?
  - Debiased because conventional variable selection is biased and this method is not
  - Double because there are two parallel steps which together remove the bias

We at RIPL have written a package in R called ‘riplDML’ that implements DML

Overview

Double/Debiased Machine Learning (DML)

- A method of controlling for a high-dimensional nuisance parameter; a variable selection method
- Is it double or debiased or both?
  - Debiased because conventional variable selection is biased and this method is not
  - Double because there are two parallel steps which together remove the bias

We at RIPL have written a package in R called 'riplDML' that implements DML

Overview

Double/Debiased Machine Learning (DML)

- A method of controlling for a high-dimensional nuisance parameter; a variable selection method
- Is it double or debiased or both?
    - Debiased because conventional variable selection is biased and this method is not
    - Double because there are two parallel steps which together remove the bias

We at RIPL have written a package in R called 'riplDML' that implements DML

Overview

Double/Debiased Machine Learning (DML)

- A method of controlling for a high-dimensional nuisance parameter; a variable selection method
- Is it double or debiased or both?
  - Debiased because conventional variable selection is biased and this method is not
  - Double because there are two parallel steps which together remove the bias

We at RIPL have written a package in R called ‘riplDML’ that implements DML

Related Literature

Base DML

- Variable selection - Belloni, Chernozhukov, and Hansen (2014)
- Full DML - Chernozhukov et al. (2018)

DML with Heterogeneous Treatment Effects

- Technical treatment - Semenova et al. (2017)
- Easy examples - Goldman and Quistorff (2018)

Variable Selection

$$y_i = d_i\beta + x_i\gamma + \epsilon_i$$
$$\mathbb{E}(\epsilon_i|d_i, x_i) = 0$$

- $y_i[1 \times 1]$ = Outcome (e.g. wages) for individual i
- $d_i[1 \times J]$ = Treatment variables (e.g., program participation) and necessary covariates (e.g., PSAT score)
- $x_i[1 \times K]$ = Additional covariates (e.g., gender)

Variable Selection

$$y_i = d_i\beta + x_i\gamma + \epsilon_i$$
$$\mathbb{E}(\epsilon_i|d_i, x_i) = 0$$

Suppose x_i is highly multidimensional

- Many distinct variables
- Many possible functional forms

OLS may be inconvenient, inefficient, expensive, or impossible to estimate. Can we select a relevant subset of covariates x_i?

Variable Selection

$$y_i = d_i\beta + x_i\gamma + \epsilon_i$$
$$\mathbb{E}(\epsilon_i | d_i, x_i) = 0$$

Suppose x_i is highly multidimensional

- Many distinct variables
- Many possible functional forms

OLS may be inconvenient, inefficient, expensive, or impossible to estimate. Can we select a relevant subset of covariates x_i?

Variable Selection

$$y_i = d_i\beta + x_i\gamma + \epsilon_i$$
$$\mathbb{E}(\epsilon_i|d_i, x_i) = 0$$

Suppose x_i is highly multidimensional

- Many distinct variables
- Many possible functional forms

OLS may be inconvenient, inefficient, expensive, or impossible to estimate. Can we select a relevant subset of covariates x_i?

Variable Selection

$$y_i = d_i\beta + x_i\gamma + \epsilon_i$$
$$\mathbb{E}(\epsilon_i | d_i, x_i) = 0$$

Suppose x_i is highly multidimensional

- Many distinct variables
- Many possible functional forms

OLS may be inconvenient, inefficient, expensive, or impossible to estimate. Can we select a relevant subset of covariates x_i?

Linear Model Decomposition

If OLS were feasible, the OLS estimator $\hat{\beta}$ could either be (a) calculated directly:

$$\text{Let } Y = \begin{bmatrix} y_1 \\ y_2 \\ ... \\ y_n \end{bmatrix}, D = \begin{bmatrix} d_1 \\ d_2 \\ ... \\ d_n \end{bmatrix}, X = \begin{bmatrix} x_1 \\ x_2 \\ ... \\ x_n \end{bmatrix}, Z = \begin{bmatrix} D & X \end{bmatrix}$$

$$\hat{\beta}^{OLS} = ((Z^T Z)^{-1} Z^T Y)_{1..K}$$

Or (b) calculated by partialing out covariates in the following decomposition:

$$\hat{D}^{OLS} = X(X^T X)^{-1} X^T D, \ \tilde{X} = D - \hat{D}^{OLS}$$
$$\hat{Y}^{OLS} = X(X^T X)^{-1} X^T Y, \ \tilde{Y} = Y - \hat{Y}^{OLS}$$
$$\hat{\beta}^{OLS} = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T \tilde{Y}$$

DML Residualizes Separately

DML substitutes other predictions of Y, D for $\hat{Y}^{OLS}$, $\hat{D}^{OLS}$:

$$\hat{D}_j^{DML} = f(D_j, X),\ \hat{Y}^{DML} = f(Y, X)$$
$$\text{Let } \tilde{D} = D - \hat{D}^{DML},\ \tilde{Y} = Y - \hat{Y}^{DML}$$
$$\hat{\beta}^{DML} = (\tilde{D}^T\tilde{D})^{-1}\tilde{D}^T\tilde{Y}$$
$$\hat{\beta}^{DML} \neq \hat{\beta}^{OLS},\ \text{but close enough!}$$
$$\hat{\beta}^{DML} \xrightarrow{p} \beta$$

where $f(\cdot,\cdot)$ can be any function such that $(f(A,X))_i \xrightarrow{p} \mathbb{E}(a_i|x_i)$; that is, any consistent method of generating predicted values from X can be used

DML Valid and Feasible

$$\tilde{D}_j^{DML} = D_j - f(D_j, X),\ \tilde{Y}^{DML} = Y - f(Y, X)$$
$$\hat{\beta}^{DML} = (\tilde{D}^T \tilde{D})^{-1} \tilde{D}^T \tilde{Y}$$
$$\hat{\beta}^{DML} \xrightarrow{p} \beta$$

DML is based on the same decomposition logic as we saw above, but it is not bound to OLS in prediction function $f(\cdot, \cdot)$:

- Can iteratively test possible variables
- Can select a subset of relevant variables
- Don't need to preserve the set of selected variables across d and y
- Don't even need to know the set of selected variables
- Widens the set of available methods (e.g., Random Forest)

DML Valid and Feasible

$$\tilde{D}_j^{DML} = D_j - f(D_j, X),\ \tilde{Y}^{DML} = Y - f(Y, X)$$
$$\hat{\beta}^{DML} = (\tilde{D}^T\tilde{D})^{-1}\tilde{D}^T\tilde{Y}$$
$$\hat{\beta}^{DML} \xrightarrow{p} \beta$$

DML is based on the same decomposition logic as we saw above, but it is not bound to OLS in prediction function $f(\cdot,\cdot)$:

- Can iteratively test possible variables
- Can select a subset of relevant variables
- Don't need to preserve the set of selected variables across d and y
- Don't even need to know the set of selected variables
- Widens the set of available methods (e.g., Random Forest)

DML Valid and Feasible

$$\tilde{D}_j^{DML} = D_j - f(D_j, X),\ \tilde{Y}^{DML} = Y - f(Y, X)$$
$$\hat{\beta}^{DML} = (\tilde{D}^T\tilde{D})^{-1}\tilde{D}^T\tilde{Y}$$
$$\hat{\beta}^{DML} \xrightarrow{p} \beta$$

DML is based on the same decomposition logic as we saw above, but it is not bound to OLS in prediction function $f(\cdot, \cdot)$:

- Can iteratively test possible variables
- Can select a subset of relevant variables
- Don't need to preserve the set of selected variables across d and y
- Don't even need to know the set of selected variables
- Widens the set of available methods (e.g., Random Forest)

DML Valid and Feasible

$$\tilde{D}_j^{DML} = D_j - f(D_j, X),\ \tilde{Y}^{DML} = Y - f(Y, X)$$
$$\hat{\beta}^{DML} = (\tilde{D}^T \tilde{D})^{-1} \tilde{D}^T \tilde{Y}$$
$$\hat{\beta}^{DML} \xrightarrow{p} \beta$$

DML is based on the same decomposition logic as we saw above, but it is not bound to OLS in prediction function $f(\cdot, \cdot)$:

- Can iteratively test possible variables
- Can select a subset of relevant variables
- Don't need to preserve the set of selected variables across d and y
- Don't even need to know the set of selected variables
- Widens the set of available methods (e.g., Random Forest)

DML Valid and Feasible

$$
\begin{aligned}
\tilde{D}_j^{DML} &= D_j - f(D_j, X),\ \tilde{Y}^{DML} = Y - f(Y, X) \\
\hat{\beta}^{DML} &= (\tilde{D}^T \tilde{D})^{-1} \tilde{D}^T \tilde{Y} \\
\hat{\beta}^{DML} &\xrightarrow{p} \beta
\end{aligned}
$$

DML is based on the same decomposition logic as we saw above, but it is not bound to OLS in prediction function $f(\cdot, \cdot)$:

- Can iteratively test possible variables
- Can select a subset of relevant variables
- Don't need to preserve the set of selected variables across d and y
- Don't even need to know the set of selected variables
- Widens the set of available methods (e.g., Random Forest)

Summary of Base DML

Step 1:
Residualize outcome y_i and treatments d_i using covariates x_i and prediction function $f(\cdot,\cdot)$:

$$\tilde{y}_i = y_i - (f(Y, X))_i$$
$$\tilde{d}_{ij} = d_{ij} - (f(D_j, X))_i$$

Step 2:
Estimate treatment effects:

$$\tilde{y}_i = \tilde{d}_i \beta + \tilde{\epsilon}$$
$$\mathbb{E}(\tilde{\epsilon}|\tilde{d}_i) \approx 0$$

Using riplDML Package For Base DML

To use base DML in R:

```
library('riplDML')
riplDML::dml.lm <- function(data, y_var, x_vars
          , d_vars = d_vars,
          , first_stage_family, predict_fun
          , second_stage_family = 'mr')
```

where the columns of 'data' named 'y_var', 'x_vars', and 'd_vars' correspond to Y, X, and D, respectively; 'first_stage_family' indicates the method of prediction used; and if desired, a user-defined prediction function may be input as 'predict_fun'.

Model with Heterogeneity by Covariates

Suppose that the effect of treatment d_{ij} varies across $(a_m(x))_m$, some (known) transformations of covariates, such that:

$$y_i = \sum_j \sum_{m \in M_j} d_{ij} a_m(x_i) \beta_{jm} + x_i \gamma + \epsilon_i$$

$$\mathbb{E}(\epsilon_i | d_i, x_i) = 0$$

This might make sense if:

- Effect of a medical treatment is stronger for women
- Personal finance course is more helpful to people with more debt

Model with Heterogeneity by Covariates

Suppose that the effect of treatment d_{ij} varies across $(a_m(x))_m$, some (known) transformations of covariates, such that:

$$y_i = \sum_j \sum_{m \in M_j} d_{ij} a_m(x_i) \beta_{jm} + x_i \gamma + \epsilon_i$$

Let $d_{ijm} = d_{ij} a_m(x_i)$. Then base DML works just fine:

$$\begin{aligned} \tilde{y}_i &= y_i - (f(Y, X))_i \\ \tilde{d}_{ijm} &= d_{ijm} - (f(D_{jm}, X))_i \\ \tilde{y}_i &= \sum_{j,m} \tilde{d}_{ijm} \beta_{jm} + \tilde{\epsilon}_i \, , \, \mathbb{E}(\tilde{\epsilon}_i | \tilde{d}_i) \approx 0 \end{aligned}$$

Simplifying Prediction of Treatment

But do we really have to estimate $\hat{d}_{ijm} = (f(D_{jm}, X))_i$ completely separately for each m? Note that:

$$\mathbb{E}(d_{ijm}|x_i) = \mathbb{E}(d_{ij}a_m(x_i)|x_i) = \mathbb{E}(d_{ij}|x_i)a_m(x_i)$$

Let our estimator for $\mathbb{E}(d_{ijm}|x_i)$ be $f_m(D_{jm}, X) = f(D_j, X)a_m(x_i)$. Then each treatment d_j is residualized once, and that prediction is used several times:

$$\begin{aligned}
\tilde{d}_{ij} &= d_{ij} - (f(D_j, X))_i \\
\tilde{d}_{ijm} &= d_{ijm} - (f_m(D_{jm}, X))_i = d_{ij}a_m(x_i) - f(D_j, X)a_m(x_i) \\
&= \tilde{d}_{ij}a_m(x_i)
\end{aligned}$$

Summary Of Heterogeneity By Covariates

Step 1:
Residualize outcome y_i and treatments d_i using covariates x_i and prediction function $f(\cdot,\cdot)$:

$$\tilde{y}_i = y_i - (f(Y, X))_i$$
$$\tilde{d}_{ij} = d_{ij} - (f(D_j, X))_i$$

Step 2:
Estimate treatment effects:

$$\tilde{y}_i = \sum_j \sum_{m \in M_j} \tilde{d}_{ij} a_m(x_i) \beta_{jm} + \tilde{\epsilon}$$
$$\mathbb{E}(\tilde{\epsilon}|\tilde{d}_i) \approx 0$$

Using riplDML Package For Heterogeneity By Covariates

```
library('riplDML')
riplDML::dml.lm <- function(data, y_var, x_vars
         , h_vars = h_vars,
         , first_stage_family, predict_fun
         , second_stage_family = 'mr')
```

where ‘h_vars’ is a matrix, in which a row corresponds to a heterogeneous treatment d_{ijm} and the columns are:

- ‘d’ = the name of the column in ‘data’ corresponding to d_{ij}
- ‘fx’ = the name of the column in ‘data’ corresponding to the function of covariates $a_m(x_i)$
- ‘fxd.name’ = the name of the corresponding coefficient estimate in regression output

Model with Heterogeneity By Treatment Type

Suppose that the treatment variables d_{ij} partition the sample. That is, each treatment is binary and the treatments are mutually exclusive, such that:

$$y_i = \sum_j d_{ij}\beta_j + x_i\gamma + \epsilon_i$$
$$z_i = \sum_j d_{ij}$$
$$d_{ij} \in \{0, 1\}\,,\; z_i \in \{0, 1\}$$
$$\mathbb{E}(\epsilon_i | (d_{ij})_j, x_i) = 0$$

This might make sense:

- Multiple workforce training programs
- Industry or occupation switching

Add A Simplifying Assumption

Further suppose that we are willing to consider a somewhat more restrictive assumption relating the treatments to the covariates. Letting $\phi_j = \frac{\mathbb{E}(d_{ij})}{\mathbb{E}(z_i)}$:

$$\mathbb{E}(d_{ij}|x_i) = \mathbb{E}(z_i|x_i)\phi_j \tag{1}$$

Under this assumption, the likelihood of any-treatment z_i may be related to covariates x_i, but the choice or assignment of a specific treatment d_{ij} within the group of treated units is unrelated to covariates x_i.

Simpler Estimators

Given (1), we may be able to use an estimator of β that only predicts and residualizes treatment variable (any-treatment z_i).

1. ($\hat{\beta}_j^{MR}$) from multiple residualizations: $\tilde{y}_i = \sum_j \tilde{d}_{ij}\beta_j + \epsilon_i$

$$\tilde{y}_i = y_i - \hat{\mathbb{E}}(y_i|x_i)$$
$$\tilde{d}_{ij} = d_{ij} - \hat{\mathbb{E}}(d_{ij}|x_i)$$

2. ($\hat{\beta}_j^{SR1}$) from single residualization (original): $\tilde{y}_i = \sum_j \tilde{d}_{ij}\beta_j + \epsilon_i$

$$\tilde{y}_i = y_i - \hat{\mathbb{E}}(y_i|x_i)$$
$$\tilde{z}_i = z_i - \hat{\mathbb{E}}(z_i|x_i)$$
$$\tilde{x}_{ij} = d_{ij}\tilde{z}_i$$

3. ($\hat{\beta}_j^{SR2}$) from single residualization (new): $\tilde{y}_i = \sum_j d_{ij}\beta_j + \hat{z}_i\alpha + \epsilon_i$

$$\tilde{y}_i = y_i - \hat{\mathbb{E}}(y_i|x_i)$$
$$\hat{z}_i = \hat{\mathbb{E}}(z_i|x_i)$$

Comparison of Estimation Strategies

Let $d_i = \begin{bmatrix} d_{i1} & d_{i2} & \dots & d_{iJ} \end{bmatrix}$ and $\beta^T = \begin{bmatrix} \beta_1 & \beta_2 & \dots & \beta_J \end{bmatrix}$

1. $\hat{\beta}^{MR} \xrightarrow{p} \beta$ details

   computationally expensive?

2. $\hat{\beta}^{SR1} \xrightarrow{p} \beta + \mathbb{E}(\tilde{d}_i^T \tilde{d}_i)^{-1} \mathbb{E}(\tilde{d}_i^T (x_i \Delta(\iota d_i - I_J)))\beta$ details

   bias term includes 3rd, 4th moments of x_i

3. $\hat{\beta}^{SR2} \xrightarrow{p}$
   $\beta + (\mathbb{E}(d_i^T d_i) - \mathbb{E}(\theta^T \hat{d}_i^T \hat{z}_i \theta))^{-1} (\mathbb{E}(\Delta^T x_i^T x_i \Delta) - \mathbb{E}(\theta^T \hat{z}_i^T \hat{z}_i \theta))\beta$
   details

   when (1) holds, bias term is zero

Using riplDML Package For Heterogeneity By Partition

To estimate $\hat{\beta}^{SR1}$ and $\hat{\beta}^{SR2}$:

```
library('riplDML')
riplDML::dml.lm <- function(data, y_var, x_vars
         , d_vars = d_vars,
         , first_stage_family, predict_fun
         , second_stage_family = 'sr1'))
riplDML::dml.lm <- function(data, y_var, x_vars
         , d_vars = d_vars,
         , first_stage_family, predict_fun
         , second_stage_family = 'sr2')
```

Conclusion

Double Machine Learning (DML) solves an important variable selection problem

- Allows us to use a large number of distinct variables
- Allows flexibility in the functional form
- Yields a consistent estimator of the object of interest
- Can plug in a variety of different machine learning methods

The R package ‘riplDML’ implements DML estimators

- One or many treatment variables
- Treatment effect heterogeneity by covariates
- Allows user to define a prediction algorithm

References I

Belloni, A., Chernozhukov, V., & Hansen, C. (2014). Inference on treatment effects after selection among high-dimensional controls. *The Review of Economic Studies*, *81*(2), 608–650.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., & Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, *21*(1), C1–C68.

Goldman, M., & Quistorff, B. (2018, June 12). Pricing engine: Estimating causal impacts in real world business settings.

Semenova, V., Goldman, M., Chernozhukov, V., & Taddy, M. (2017). Estimation and inference on heterogeneous treatment effects in high-dimensional dynamic panels [Publisher: arXiv Version Number: 4].

Multiple Residualization Bias

$$
\begin{aligned}
\hat{\beta}^{MR} &\xrightarrow{p} \mathbb{E}((d_i - x_i\delta)^T(d_i - x_i\Delta))^{-1}\mathbb{E}((d_i - x_i\Delta)^T(y_i - x_i(\gamma + \Delta\beta))) \\
&\xrightarrow{p} \mathbb{E}(\eta_i^T\eta_i)^{-1}\mathbb{E}(\eta_i^T(\eta_i\beta + (x_i\gamma - \gamma))) \\
&\xrightarrow{p} \beta
\end{aligned}
$$

back

Single Residualization Bias - SR1

Let $\iota : [J \times 1]$ such that each element of ι is equal to 1, and let I be an identity matrix of size J. Also, let $\psi_{ij} = x_i \Delta_j$ and let $\psi_i = x_i \Delta$. Then:

$$\begin{aligned}
\tilde{x}_{ij} &\xrightarrow{p} (d_i \iota - x_i \Delta \iota) d_{ij} \, , \, \tilde{x}_i \xrightarrow{p} (d_i \iota - x_i \Delta \iota) d_i = d_i - x_i \Delta \iota d_i \\
\tilde{y}_i &\xrightarrow{p} (d_i - x_i \Delta \iota d_i)\beta + x_i(\gamma + \Delta \iota d_i \beta - \gamma - \Delta \beta) + \epsilon_i \\
&\xrightarrow{p} \tilde{x}_i \beta + x_i(\Delta(\iota d_i - I)\beta) + \epsilon_i \\
\hat{\beta}^{SR1} &\xrightarrow{p} \mathbb{E}(\tilde{x}_i^T \tilde{x}_i)^{-1} \mathbb{E}(\tilde{x}_i^T (\tilde{x}_i \beta + x_i \Delta(\iota d_i - I)\beta)) \\
\hat{\beta}_j^{SR1} &\xrightarrow{p} \beta_j + \frac{\mathbb{E}((1 - \psi_i \iota)(\psi_{ij})(\psi_i \iota \beta_j - \psi_{ij} \psi_i \beta))}{\mathbb{E}((1 - \psi_i \iota)^2 \psi_{ij})}
\end{aligned}$$

back

Single Residualization Bias - SR2 (1/2)

0. Let $\iota : [J \times 1]$ such that each element of ι is equal to 1, and let $z_i = d_i\iota = x_i\Delta\iota + \eta_i\iota$. Also, let $\delta = \Delta\iota$, let $\hat{\delta} = \mathbb{E}(x_i^T x_i)^{-1}\mathbb{E}(x_i^T z_i)$, let $\hat{z}_i = x_i\hat{\delta}$, and let $\theta = \mathbb{E}(\hat{z}_i^T \hat{z}_i)^{-1}\mathbb{E}(\hat{z}_i^T d_i)$. Then:

$$
\begin{aligned}
\tilde{y}_i &\xrightarrow{p} d_i\beta + x_i(\gamma - \gamma - \Delta\beta) + \epsilon_i = d_i\beta - x_i\Delta\beta + \hat{z}_i\theta\beta - \hat{z}_i\theta\beta + \epsilon_i \\
&\xrightarrow{p} (d_i - \hat{z}_i\theta)\beta + (\hat{z}_i\theta\beta - x_i\Delta\beta) + \epsilon_i \\
\mathbb{E}(\hat{z}_i^T \tilde{y}_i) &\xrightarrow{p} \mathbb{E}(\iota^T\Delta^T x_i^T \tilde{y}_i) = 0 \\
\mathbb{E}(d_i^T \hat{z}_i\theta) &= \mathbb{E}(x_i^T \hat{z}_i\theta) = \mathbb{E}(\theta^T \hat{z}_i\hat{z}_i\theta) \\
\mathbb{E}(d_i^T x_i\Delta) &= \mathbb{E}(\Delta^T x_i^T x_i\Delta) \\
\hat{\beta}^{SR2} &\xrightarrow{p} \mathbb{E}((d_i - z_i\theta)^T(d_i - z_i\theta))^{-1}\mathbb{E}((d_i - z_i\theta)^T\tilde{y}_i)
\end{aligned}
$$

back

Single Residualization Bias - SR2 (2/2)

$$\begin{aligned}
\hat{\beta}^{SR2} &\xrightarrow{p} \mathbb{E}((d_i - z_i\theta)^T(d_i - z_i\theta))^{-1}\mathbb{E}((d_i - z_i\theta)^T\tilde{y}_i) \\
&\xrightarrow{p} \beta + (\mathbb{E}(d_i^T d_i) - \mathbb{E}(\theta^T \hat{z}_i^T \hat{z}_i\theta))^{-1}\mathbb{E}((d_i - \hat{z}_i\theta)^T(\hat{z}_i\theta\beta - x_i\Delta\beta)) \\
&\xrightarrow{p} \beta - (\mathbb{E}(d_i^T d_i) - \mathbb{E}(\theta^T \hat{z}_i^T \hat{z}_i\theta))^{-1}\mathbb{E}((d_i - \hat{z}_i\theta)^T x_i\Delta)\beta \\
&\xrightarrow{p} \beta - (\mathbb{E}(d_i^T d_i) - \mathbb{E}(\theta^T \hat{z}_i^T \hat{z}_i\theta))^{-1}(\mathbb{E}(\Delta^T x_i^T x_i\Delta) - \mathbb{E}(\theta^T \hat{z}_i^T \hat{z}_i\theta))\beta
\end{aligned}$$

So that the bias term approaches zero when $\hat{z}_i$ predicts d_i as well as x_i predicts d_i.

back