Quant II

Machine Learning and External Validity

Ye Wang

5/6/2020

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?
- Variable creation: train a model and use it to code data

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?
- Variable creation: train a model and use it to code data
- Predicting nuisance parameters

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?
- Variable creation: train a model and use it to code data
- Predicting nuisance parameters
- Estimating heterogeneous treatment effects and generate the optimal assignment

Variable creation

- The problem: transform texts, images, audio, or even video into variables

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)
- Example IV: audio and video processing (Knox and Lucas, 2018)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)
- Example IV: audio and video processing (Knox and Lucas, 2018)
- Example V: identify Russian bots on Twitter (Stukal et al., 2017)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)
- Example IV: audio and video processing (Knox and Lucas, 2018)
- Example V: identify Russian bots on Twitter (Stukal et al., 2017)
- What if it is too expensive? Active learning (Miller et al., 2019)

Experimental analysis

- Experiment $+$ gradient descent (Wager and Kuang, 2019)

Experimental analysis

- Experiment + gradient descent (Wager and Kuang, 2019)
- What is the optimal price offered to drivers for UBER?

Experimental analysis

- Experiment + gradient descent (Wager and Kuang, 2019)
- What is the optimal price offered to drivers for UBER?
- Drivers decide whether to work after observing the price (an inverted U-shape profit curve).

Experimental analysis

- Experiment + gradient descent (Wager and Kuang, 2019)
- What is the optimal price offered to drivers for UBER?
- Drivers decide whether to work after observing the price (an inverted U-shape profit curve).
- Assign each driver $p_i = p_0 + \varepsilon_i$.

Experimental analysis

- Experiment + gradient descent (Wager and Kuang, 2019)
- What is the optimal price offered to drivers for UBER?
- Drivers decide whether to work after observing the price (an inverted U-shape profit curve).
- Assign each driver $p_i = p_0 + \varepsilon_i$.
- Calculate the slope of tangent at p_0, and do gradient descent to find p_1.

Experimental analysis

- Experiment + gradient descent (Wager and Kuang, 2019)
- What is the optimal price offered to drivers for UBER?
- Drivers decide whether to work after observing the price (an inverted U-shape profit curve).
- Assign each driver $p_i = p_0 + \varepsilon_i$.
- Calculate the slope of tangent at p_0, and do gradient descent to find p_1.
- Repeat until convergence.

Methods: predicting nuisance parameters

- Some relationships in causal inference can be non-causal.
- We just need to fit/predict it with a high accuracy.
  - Example I: Propensity score
  - Example II: First stage of IV
  - Example III: Response surface (what covariates to control for)
- These are "nuisance parameters" that have no causal interpretation.

Methods: predicting nuisance parameters

- Directly applying machine learning algorithms to estimate nuisance parameters leads to severe bias in finite sample.

Methods: predicting nuisance parameters

- Directly applying machine learning algorithms to estimate nuisance parameters leads to severe bias in finite sample.
- Cattaneo et al. (2018): two-stage estimation with too many covariates in the first stage is biased.
- Belloni et al. (2013) show that we need run "double selection" to obtain satisfying results.
- One model for the outcome, and the other for the treatment.

Methods: predicting nuisance parameters

- Directly applying machine learning algorithms to estimate nuisance parameters leads to severe bias in finite sample.
- Cattaneo et al. (2018): two-stage estimation with too many covariates in the first stage is biased.
- Belloni et al. (2013) show that we need run "double selection" to obtain satisfying results.
- One model for the outcome, and the other for the treatment.
- But why?

Double machine learning

- Let's consider the following DGP:

$$Y_i = \theta D_i + g_0(\mathbf{X}_i) + U_i$$

$$D_i = m_0(\mathbf{X}_i) + V_i$$

- We have $D_i \perp \{Y_i(1), Y_i(0)\} | \mathbf{X}_i$.

Double machine learning

- The classic model-based approach will find an estimate $\hat{g}$ for g_0.
- Then,

$$\hat{\theta} = \frac{\sum_{i=1}^{N} D_i(Y_i - \hat{g}(\mathbf{X}_i))}{N_1} - \frac{\sum_{i=1}^{N}(1 - D_i)(Y_i - \hat{g}(\mathbf{X}_i))}{N_0}$$

Double machine learning

- The classic model-based approach will find an estimate $\hat{g}$ for g_0.
- Then,

$$\hat{\theta} = \frac{\sum_{i=1}^{N} D_i(Y_i - \hat{g}(\mathbf{X}_i))}{N_1} - \frac{\sum_{i=1}^{N}(1 - D_i)(Y_i - \hat{g}(\mathbf{X}_i))}{N_0}$$

- This is "single selection."

Double machine learning

- Nevertheless, we can show that:

$$
\begin{aligned}
&\sqrt{N}(\hat{\theta} - \theta) \\
=&\sqrt{N}\left[\frac{\sum_{i=1}^{N} D_i U_i}{N_1} - \frac{\sum_{i=1}^{N}(1 - D_i)U_i}{N_0}\right] \\
+&\sqrt{N}\left[\frac{\sum_{i=1}^{N} D_i(g_0(\mathbf{X}_i) - \hat{g}(\mathbf{X}_i))}{N_1} - \frac{\sum_{i=1}^{N}(1 - D_i)(g_0(\mathbf{X}_i) - \hat{g}(\mathbf{X}_i))}{N_0}\right]
\end{aligned}
$$

- The first part is just the Hajek estimator, which converges to $N(0, I)$.
- But the second part may diverge to infinity as the convergence of $\hat{g}$ to g_0 is often slow.

Double machine learning

- Denote $E[Y_i|\mathbf{X}_i] = m_0(\mathbf{X}_i)\theta + g_0(\mathbf{X}_i)$ as $l_0(\mathbf{X}_i)$.
- We use machine learning to estimate $m_0(\mathbf{X}_i)$ and $l_0(\mathbf{X}_i)$.
- Then, we take the residual: $\hat{V}_i = D_i - \hat{m}(\mathbf{X}_i)$ and $\hat{W}_i = Y_i - \hat{l}(\mathbf{X}_i)$.
- Finally, $\hat{\theta}$ is estimated by regressing $\hat{W}_i$ on $\hat{V}_i$.

Double machine learning

- Denote $E[Y_i|\mathbf{X}_i] = m_0(\mathbf{X}_i)\theta + g_0(\mathbf{X}_i)$ as $l_0(\mathbf{X}_i)$.
- We use machine learning to estimate $m_0(\mathbf{X}_i)$ and $l_0(\mathbf{X}_i)$.
- Then, we take the residual: $\hat{V}_i = D_i - \hat{m}(\mathbf{X}_i)$ and $\hat{W}_i = Y_i - \hat{l}(\mathbf{X}_i)$.
- Finally, $\hat{\theta}$ is estimated by regressing $\hat{W}_i$ on $\hat{V}_i$.
- Intuitively, the second part of the bias is now decided by $(\hat{m}(\mathbf{X}_i) - m_0(\mathbf{X}_i))(\hat{l}(\mathbf{X}_i) - l_0(\mathbf{X}_i))$ plus $V_i(\hat{g}(\mathbf{X}_i) - g_0(\mathbf{X}_i))$.
- Even when each estimator converges to the true value slowly, their product may have a satisfying convergence rate.

Double machine learning

- This is called “Robinson’s Transformation” (Robinson, 1988).
- The transformation allows us to achieve “Neyman orthogonality,” meaning the bias from estimating nuisance parameters have negligible influence on the estimation of causal parameters.

Double machine learning

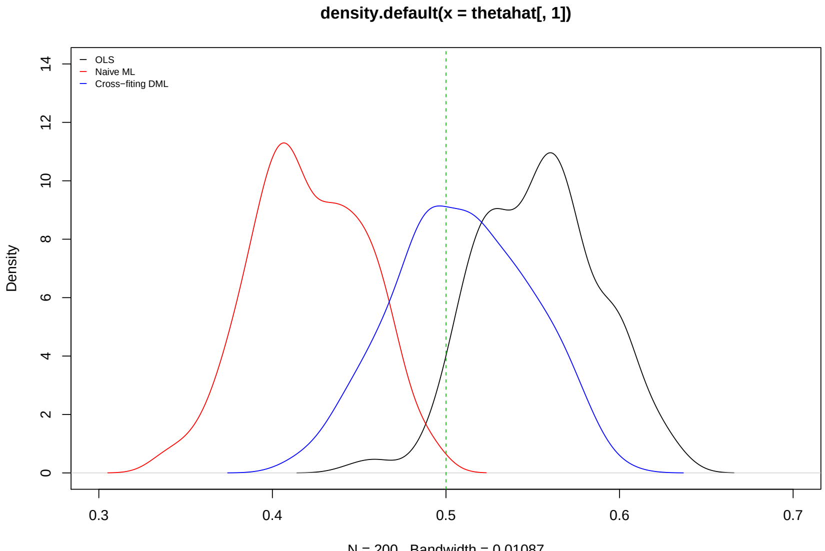
- This is called "Robinson's Transformation" (Robinson, 1988).
- The transformation allows us to achieve "Neyman orthogonality," meaning the bias from estimating nuisance parameters have negligible influence on the estimation of causal parameters.
- Double machine learning is built upon the same idea as the doubly robust estimator.
- You need to get either the response surface or the propensity score correct, and you have higher efficiency by getting both correct.

Double machine learning

- We still have a remainder: $V_i(\hat{g}(\mathbf{X}_i) - g_0(\mathbf{X}_i))$.
- This term only relies on the property of $\hat{g}$.
- If you use LASSO, the remainder converges to zero at a fast rate.
- For more general algorithms, we use sample splitting to eliminate it.
- As $\hat{g}$ is generated on an independent sample, it should be orthogonal to V_i.
- We can split the sample multiple times and take the average over the estimates.
- There is no efficiency loss.

Double machine learning

```
##           OLS       Naive ML Cross-fiting DML
##     0.5532314      0.4208227        0.5089906
```



Double machine learning

- Belloni et al. (2012): use LASSO/Post-LASSO to select instruments.
- Belloni et al. (2013): use LASSO/Post-LASSO to select covariates.
- Chernozhukov et al. (2016): use LASSO/Post-LASSO to select covariates in panel data.
- Belloni et al. (2016): use double machine learning to to estimate any functional.
- Chernozhukov et al. (2018): use double machine learning to estimate nuisance parameters.

The End

Good luck with your final!