SOC 690S: Machine Learning in Causal Inference

Week 2: Machine Learning Basics

Wenhao Jiang
Department of Sociology, Fall 2025

Motivation and High-Dimensional Data

Sample Sandwich Estimator Fails in High Dimension

- Last week, we discussed the problem of *sample* linear regression in the high-dimensional regime ($p/n \not\to 0$)
- Even if the true *data-generating process* (DGP) in the *population* is correctly specified ($E[\hat{\beta}] = \beta$), high dimensionality causes problems for the sample regression
- In particular, the variance of the OLS (sandwich) estimator is *underestimated* and inconsistent at rate $\sqrt{p/n}$, because the *sample* covariance matrix no longer converges to its *population* counterpart—the *Law of Large Numbers* fails under high-dimensional regime

$$\left\| \frac{1}{n} \sum_{i=1}^{n} X_i X_i' - E[X_i X_i'] \right\|_{\text{op}} \sim O_p\left(\sqrt{\frac{p}{n}}\right)$$

Sample Sandwich Estimator is Inconsistent

- Indeed, in high-dimensional regimes, $\hat{\beta}$ is no longer $\sqrt{n}$-consistent

$$\sqrt{n}(\hat{\beta} - \beta) \overset{d}{\nrightarrow} \mathcal{N}\left(0,\ E[X_i X_i']^{-1}\, E[e_i^2 X_i X_i']\, E[X_i X_i']^{-1}\right)$$

- The estimation uncertainty does not vanish to 0 even if $n \to \infty$ when $p/n \nrightarrow 0$
  - The number of parameters grows with n
  - The overall estimation uncertainty does not vanish
  - $\hat{\beta}$ is unbiased but not consistent

Out-of-Sample Prediction is Poor in High Dimension

- Another perspective: the *out-of-sample* prediction error relative to the true regression does not converge to 0 when $n \to \infty$ in high dimension

Out-of-Sample Prediction is Poor in High Dimension

- Another perspective: the *out-of-sample* prediction error relative to the true regression does not converge to 0 when $n \to \infty$ in high dimension
- Suppose $\hat{\beta}$ is the OLS estimate, $E_X[\cdot]$ denotes averaging over fresh test samples from the population. The *root mean square prediction error* (RMSE) satisfies the high-probability bound

$$\sqrt{E_X\left[(X_i'\beta - X_i'\hat{\beta})^2\right]} \lesssim \mathrm{const}_\alpha \sqrt{E[e_i^2]}\sqrt{\frac{p}{n}}$$

- the inequality holds with probability approaching $1-\alpha$ as $n \to \infty$, where const_α is a constant that depends on the distribution (Y, X) and α (see details on page 19 in CML *Theorem 1.2.1*)

Out-of-Sample Prediction is Poor in High Dimension

- Another perspective: the *out-of-sample* prediction error relative to the true regression does not converge to 0 when $n \to \infty$ in high dimension
- Suppose $\hat{\beta}$ is the OLS estimate, $E_X[\cdot]$ denotes averaging over fresh test samples from the population. The *root mean square prediction error* (RMSE) satisfies the high-probability bound

$$\sqrt{E_X\left[(X_i'\beta - X_i'\hat{\beta})^2\right]} \lesssim \text{const}_\alpha \sqrt{E[e_i^2]}\sqrt{\frac{p}{n}}$$

- the inequality holds with probability approaching $1-\alpha$ as $n \to \infty$, where const_α is a constant that depends on the distribution (Y, X) and α (see details on page 19 in CML *Theorem 1.2.1*)
- In the low-dimensional case ($p/n \to 0$), RMSE $\to 0$
- In the high-dimensional case ($p/n \not\to 0$), the RMSE plateaus at a positive constant even as $n \to \infty$

Motivation of Dimension Reduction

- This week, we introduce basic *Machine Learning* (ML) methods to improve the prediction of $X_i'\hat{\beta}$ relative to $X_i'\beta$ (and thus Y_i) in high-dimensional data
  - Why do we care about *prediction* for new unseen data
  - It measures the extent to which the conclusion drawn from the sample is *generalizable*

Motivation of Dimension Reduction

- This week, we introduce basic *Machine Learning* (ML) methods to improve the prediction of $X_i'\hat{\beta}$ relative to $X_i'\beta$ (and thus Y_i) in high-dimensional data
  - Why do we care about *prediction* for new unseen data
  - It measures the extent to which the conclusion drawn from the sample is *generalizable*
- The most straightforward strategy is to reduce dimension, *i.e.*, the number of covariates (p) in linear regression, without losing important “information”
- This is *not* about causal inference in the framework of *potential outcome* yet, but using *sample* linear regression to approximate the DGP and the parameters (under strong *ignorability* assumption)

Practical Reason of High Dimension I

- This curse of high dimension is not rare even if **I.** the DGP is correctly specified in the *sample*
  - In cross-country analysis of economic growth, there are many country-level characteristics that may significantly predict growth ($p > n$)
  - In hard-to-reach population, researchers may want to collect as much individual-level information as possible (n is limited, while p may be large)

Practical Reason of High Dimension II

- High dimensionality may also arise when **II.** the data have large dimensional features—many covariates are available for use as regressors
  - These features may not appear in DGP, but they may approximate many unobserved characteristics in DGP that we want to model
  - In modeling the relationship between demand and price of a rare product (n is not large in reality), we want to use as many information as possible, including using textual or image features in product description, even if some of them are just noises
  - Including many regressors without selection risk high *colinearity*

Practical Reason of High Dimension III

- High dimensionality may also arise when **III.** we want to allow more flexible and interactive relationship between regressors
  - In modeling the relationship between gender and wage, we want to allow years of experience to have "non-linear" effects, years of education to be interacted with geographic indicators, etc.
  - These "non-linear" features are called *constructed* features or *transformations*

$$X = T(W) = (T_1(W), T_2(W), \ldots, T_p(W))'$$

  - *Transformations* risk *overfitting* the sample

Least Absolute Shrinkage and Selection Operator (LASSO) as a Feature Selection Method

LASSO Regression Basic Setup

- We consider a linear regression model

$$Y_i = X_i'\beta + e_i = \sum_{j=1}^{p} \beta_j X_{ij} + e_i, \quad e_i \perp X_i$$

  where p is possibly much larger than n
- Classic OLS fails in these high-dimensional settings
  - *Out-of-sample* prediction error can be very high
  - Model identification fails when the covariate matrix $rank > n$, or perfectly fits the sample data when $rank = n$

LASSO Regression Basic Setup

- For simplicity, we assume regressors are centered and normalized (*standardized*), such that β_j are on the same scale (standard packages typically do this by default)

$$E[X_{ij}^2] = 1, \quad \text{as } V(X_{ij}^2) = 1, E[X_{ij}] = 0$$

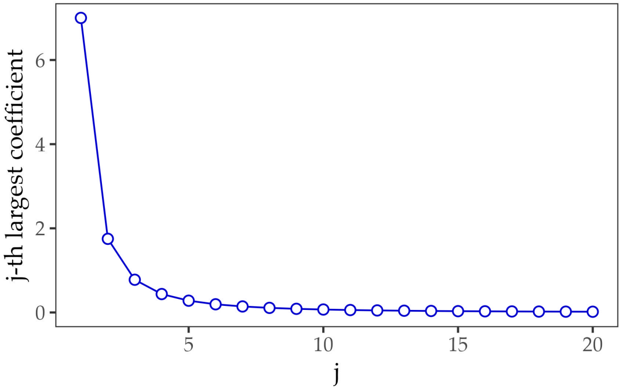
- We assume *population*-level DGP follows *approximate sparsity*[1]
  - There is a small group of regressors with relatively large coefficients whose use alone suffices to approximate the BLP $X_i'\beta$
  - The rest of the regressors are assumed to have relatively small coefficients and contribute little to the approximation of the BLP

[1]Formally, the sorted absolute values of the coefficients decay quickly. The j^{th} largest coefficient (in absolute value) denoted by $|\beta|_j$ obeys $|\beta|_j \leq Aj^{-a}, a > 1/2$ for each j

LASSO Regression Basic Setup: Approximate Sparsity

- A classic example of *approximate sparsity* is captured by regression coefficients of the form

$$\beta_j \propto 1/j^2, \quad j = 1, \ldots, p$$



LASSO Regression

- LASSO constructs $\hat{\beta}$ as the solution of the following *penalized* least squares problem

$$\hat{\beta} = \arg\min_{b \in \mathbb{R}^p} \sum_i^n (Y_i - X_i' b)^2 + \lambda \cdot \sum_{j=1}^p |b_j| \hat{\psi}_j$$

- The first term is the *prediction* error
- The the second term is called a *penalty term*, with *penalty level* λ and *penalty loading* $\hat{\psi}_j$
- The *penalty loading* is typically set as

$$\hat{\psi}_j = \sqrt{\mathbb{E}_n[X_{ji}^2]}$$

which is about 1 under *standardization*; we will omit it in the following analysis

LASSO Regression Heuristics

- The loss function can be viewed as a *trade-off* between *in-sample fit* with the measure of *complexity*

$$\mathcal{L} = \sum_{i}^{n}(Y_i - X_i'b)^2 + \lambda \cdot \sum_{j=1}^{p}|b_j|$$

- When $\lambda > 0$ (the typical setup), LASSO includes a regressor X_{ji} only if its marginal predictive ability is higher than the marginal cost $\lambda \cdot |\hat{\beta}_j|$ (ℓ_1 kink)
- Setting a larger (smaller) λ will exclude more (fewer) regressors

LASSO Regression and Penalty Choice

- Taking the derivative of the loss function with respect to $\hat{\beta}_j$

$$\mathcal{L} = \sum_{i}^{n} (Y_i - X_i' b)^2 + \lambda \cdot \sum_{j=1}^{p} |b_j|$$

$$\frac{\partial \mathcal{L}}{\partial \hat{\beta}_j} = -\hat{S}_j + \lambda \cdot \partial |\hat{\beta}_j| \text{ where } \hat{S}_j = 2 \sum_{i=1}^{n} \left(Y_i - X_i' \beta \right) X_{ji}$$

- $\partial |\hat{\beta}_j|$ has a *kink* at $\hat{\beta}_j = 0$; instead of ordinary derivatives, we use subgradients defined in convex optimization[2]

$$\partial |\hat{\beta}_j| \Big|_{\hat{\beta}_j = 0} \in [-1, 1]$$

[2] In convex optimization, local optimal is also the global optimal.

LASSO Regression and Penalty Choice

- The subgradient condition is satisfied at $\hat{\beta}_j = 0$ if

$$|\hat{S}_j| \leq \lambda \text{ where } \hat{S}_j = 2\sum_{i=1}^{n}\left(Y_i - X_i'\beta\right)X_{ji}$$

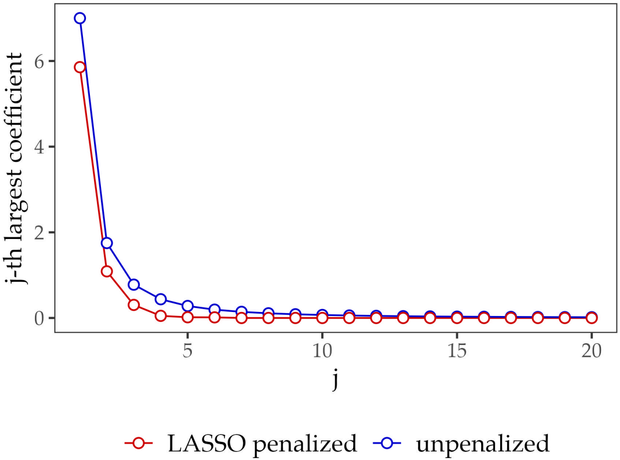
- Otherwise, the solution lies at $\hat{\beta}_j \neq 0$, found by solving the FOC

$$\hat{\beta}_j = \begin{cases} \dfrac{\hat{S}_j - \lambda}{2\,D_j}, & \text{if } \hat{S}_j > \lambda \\ \dfrac{\hat{S}_j + \lambda}{2\,D_j} & \text{if } \hat{S}_j < -\lambda \end{cases}$$

under *standardization*

$$D_j = \sum_{i=1}^{n} X_{ji}^2 \approx n$$

LASSO Regression Coefficients[3]



[3] $Y_i = X_i'\beta + e_i$, $X_i \sim \mathcal{U}(-0.5, 0.5)$, $e_i \sim \mathcal{N}(0,1)$, $\beta_j = 7/j^2$; $n = 100$, $p = 400$

LASSO Regression Caveats

- The estimates shrink towards zero relative to the unpenalized regression; this is referred as *shrinkage bias* or *regularization bias*
- LASSO estimates are therefore (slightly) biased, by design, under *approximate sparsity*

LASSO Regression Caveats

- The estimates shrink towards zero relative to the unpenalized regression; this is referred as *shrinkage bias* or *regularization bias*
- LASSO estimates are therefore (slightly) biased, by design, under *approximate sparsity*
- LASSO will not generally select the “right” set of variables
- Instead, LASSO will tend to exclude variables with small, but non-zero population coefficients
- LASSO will tend to fail to select the right variables in settings where the X_i variables are corrected

LASSO Regression Caveats

- For example, consider a scenario where variable X_1 has coefficient $\beta_1 = 0$ but is highly correlated to variables $X_2, \ldots, X_k$ that have non-zero coefficients
- It is plausible that the marginal predictive benefits of including X_1 in the model is very high when $X_2, \ldots, X_k$ are not in the model, while the marginal predictive benefits of any one of $X_2, \ldots, X_k$ is relatively low
- In this case, X_1 may enter the LASSO solution with a non-zero coefficient, while all of $X_2, \ldots, X_k$ are excluded
- This inability to select *exactly* the right regressors is not special to LASSO but shared by all variable selection procedures

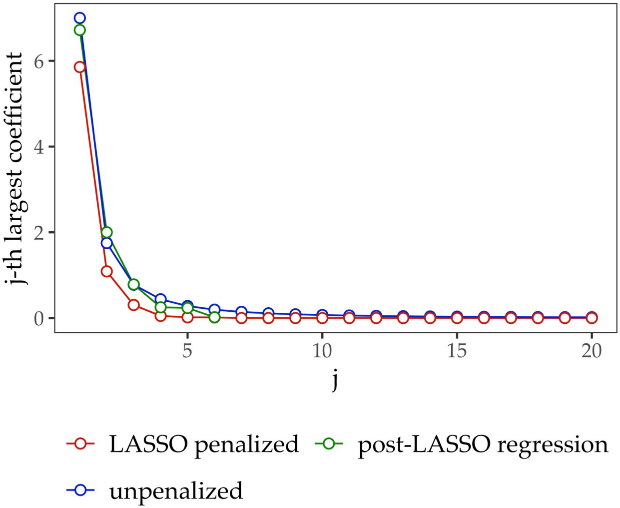
Post-LASSO Regression

- One way to adjust for the *shrinkage bias* of LASSO is to refit the OLS model using the regressors whose LASSO coefficient estimates are non-zero
- This method is called "least squares post LASSO", or simply *Post-LASSO*
- Correcting for the *shrinkage* towards zero from the non-zero coefficients sometimes delivers improvements in predictive performance

$$\hat{\beta}_{post} = \arg\min_{b \in \mathbb{R}^p} \sum_i (Y_i - X_i' b)^2 \text{ such that } b_j = 0 \text{ if } \hat{\beta}_j = 0$$

where $\hat{\beta}$ is the LASSO coefficient estimator

Post-LASSO Regression



Predictive Performance of LASSO and Post-LASSO

- Recall that, *without* feature selection, *out-of-sample* prediction from OLS estimates are poor in high-dimensional regime

$$\sqrt{E_X\left[(X_i'\beta - X_i'\hat{\beta})^2\right]} \lesssim \text{const}_\alpha \sqrt{E[e_i^2]}\sqrt{\frac{p}{n}}$$

- Intuitively, by reducing dimension p to s (*effective dimension*) using LASSO, *out-of-sample* prediction improves

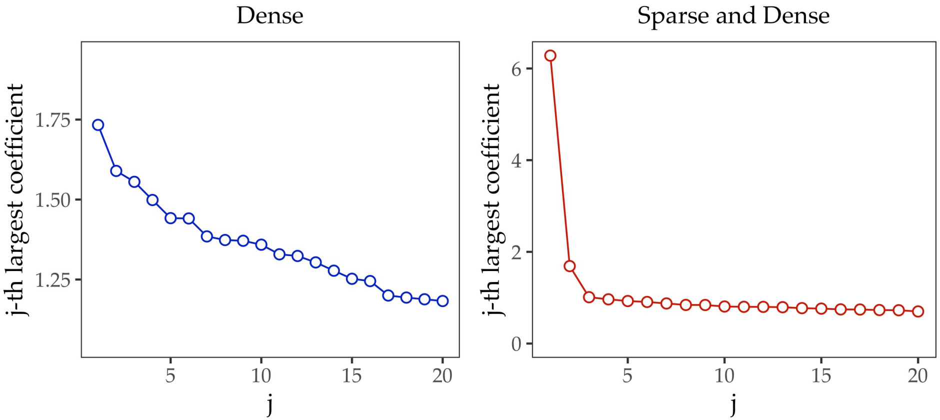
$$\sqrt{E_X\left[(X_i'\beta - X_i'\hat{\beta})^2\right]} \lesssim \text{const}_\alpha \sqrt{E[e_i^2]}\sqrt{\tfrac{s}{n}\log(max\{p,n\})}$$

Other Penalized Regression Methods beyond LASSO

Other Penalized Regression Methods beyond LASSO

- LASSO performs well in *out-of-sample* prediction when *population*-level DGP follows *approximate sparsity*
- Other DGPs exist; for example, a *dense* coefficient vector may have the vast majority or all coefficients non-zero and of *comparable* magnitude
- A *sparse and dense* structure has the vast majority of coefficients being non-zero and of similar magnitude along with a small number of relatively large coefficients

Other Penalized Regression Methods beyond LASSO



Ridge Regression for Dense Coefficients

- While LASSO performs best in an *approximately sparse* setting
- Ridge method performs best in the *dense* setting

$$\hat{\beta} = \arg\min_{b \in \mathbb{R}^p} \sum_{i}^{n} (Y_i - X_i' b)^2 + \lambda \cdot \sum_{j=1}^{p} b_j^2$$

- The latter penalty term is called ℓ_2 sphere
- In contrast to LASSO, Ridge penalizes the large values of coefficients much more aggressively and small values much less aggressively (to approximate the *dense* DGP)

Ridge Regression for Dense Coefficients

- Ridge does not set estimated coefficients to zero and does not do variable selection
- In matrix form

$$\hat{\beta}_{Ridge} = (X'X + \lambda I_p)^{-1} X'y$$

- Even if $X'X$ is singular (when $p > n$), adding λI_p makes it strictly positive definite and thus invertible
- The ridge solution is unique and numerically stable even in high dimension

Elastic Net for Sparse *or* Dense Coefficients

- Ridge and LASSO can be combined and perform well in either *sparse* or *dense* settings
- One popular hybrid is the Elastic Net with appropriate *tuning* of λ

$$\hat{\beta}_{Elastic} = \arg\min_{b \in \mathbb{R}^p} \sum_{i}^{n} (Y_i - X_i' b)^2 + \lambda_1 \cdot \sum_{j=1}^{p} b_j^2 + \lambda_2 \cdot \sum_{j=1}^{p} |b_j|$$

- By selecting different values of penalty levels λ_1 and λ_2, we have more flexibility with Elastic Net for building a good prediction rule than with just Ridge or LASSO
- The Elastic Net performs variable selection unless we completely shut down the LASSO penalty by setting $\lambda_2 = 0$

Lava for Sparse *and* Dense Coefficients

- Ridge and LASSO can also be combined and perform well in *sparse and dense* settings
- One such hybrid is the Lava method with appropriate *tuning* of λ

$$\hat{\beta}_{Lava} = \underset{b:b=\delta+\xi\in\mathbb{R}^p}{\arg\min} \sum_{i}^{n}(Y_i - X_i'b)^2 + \lambda_1 \cdot \sum_{j=1}^{p} \delta_j^2 + \lambda_2 \cdot \sum_{j=1}^{p} |\xi_j|$$

- Here components of the parameter vector are split into a "dense part" δ_j and "sparse part" ξ_j
- The minimization program automatically determines the best split into the dense and sparse parts

High-Dimensional Linear Model Simulation

- I simulate three high-dimensional ($n = 100$, $p = 400$) scenarios, where the coefficients in *population* DGP is *dense*, *sparse*, and *dense and sparse*
- I *sample* from the *population*, and evaluate the *out-of-sample* prediction by R^2

Table: *Out-of-Sample* R^2 in Simulation Experiment

Model	Sparse	Dense	Dense and Sparse
Lasso (Cross-Validation)	0.773	0.004	0.318
Lasso (Plug-in)	0.775	-0.028	0.329
Post-Lasso (Plug-in)	0.800	0.000	0.285
Ridge (Cross-Validation)	0.097	0.170	0.116
Elastic Net	0.741	0.005	0.319
Lava	0.770	0.159	0.399

How to Tune λ: Cross-Validation

- We want a valid choice of *penalty level* λ in these penalized models
- Closed-form solution is not always possible
- A convenient and theoretically valid choice can be derived from *Cross-Validation* (CV)

How to Tune λ: Cross-Validation

- We want a valid choice of *penalty level* λ in these penalized models
- Closed-form solution is not always possible
- A convenient and theoretically valid choice can be derived from *Cross-Validation* (CV)
- Remember our final goal is to find a better *prediction* model after penalizing or selecting regressors under λ
- Intuitively, we can simulate such *prediction within* the existing "training" sample *without* any test sample

Cross-Validation in Words

- We partition the *sample* data into K blocks called "folds." For example, with $K = 5$, we randomly split the data into 5 non-overlapping blocks.
- Leave one block out. Fit a prediction rule on all the other blocks. Predict the outcome observations in the left out block, and record the empirical *Mean Squared Prediction Error* (MSE). Repeat this for each block.
- Average the empirical MSEs over blocks.
- We do these steps for several or many values of the tuning parameters and choose the value of the tuning parameter that minimized the average MSEs.

Cross-Validation Formal Description

- Randomly partition the observation indices $1, \ldots, n$ into K folds $B_1, \ldots, B_K$
- For each $k = 1, \ldots, K$, fit a prediction rule denoted by $\hat{f}^{-k}(\cdot;\theta)$, where θ denotes the tuning parameters (*penalty level* λ in our case) and $\hat{f}^{-k}(\cdot;\theta)$ depends only on observations with indices not in the fold B_k
- For each $k = 1, \ldots, K$, the empirical *out-of-sample* MSE for the block B_k is

  $MSE_k(\theta) = \frac{1}{m_k} \sum_{i \in B_k} \left(Y_i - \hat{f}^{-k}(X_i;\theta) \right)^2$ where m_k is the size of the block B_k
- Compute the cross-validated MSE as

$$CV - MSE(\theta) = \frac{1}{K} \sum_{k=1}^{K} MSE_k(\theta)$$

- Choose the tuning parameter θ as a minimizer of $CV - MSE(\theta)$

How to Tune λ: Plug-in Method for LASSO

- Remember in LASSO, The subgradient condition is satisfied at $\hat{\beta}_j = 0$ if

$$|\hat{S}_j| \leq \lambda \text{ where } \hat{S}_j = 2\sum_{i=1}^{n} X_{ji}\left(Y_i - X_i'\beta\right) = 2\sum_{i=1}^{n} X_{ji}e_i$$

$$\mathbb{E}_n[\hat{S}_j] = 0, \quad \hat{V}(\hat{S}_j) \approx 4n\sigma^2$$

- By high-dimensional CLT

$$\frac{\hat{S}_j}{2\sqrt{n}\sigma} \xrightarrow{d} \mathcal{N}(0,1)$$

- We want a λ that can check each $j \in \{1, \ldots, p\}$; a theoretically valid λ is

$$\lambda = 2c\sigma\sqrt{n}z_{1-a/2p}$$

where $1-a$ is a confidence level (with $2p$ adjustments), $c = 1.1$ that practically works

How to Tune λ: Plug-in Method for LASSO

- a theoretically valid λ is

$$\lambda = 2c\sigma\sqrt{n}z_{1-a/2p}$$

- σ can be estimated from iterative method
- Let X_i^0 be a small set of regressors (a trivial choice is just the intercept); fit an unadjusted OLS and find $\hat{\beta}^0$; we define

$$\hat{\sigma}^0 := \sqrt{\mathbb{E}_n[Y_i - X_i^0\hat{\beta}^0]}$$

- Compute λ using plug-in method based on $\hat{\sigma}^0$ and LASSO estimator $\hat{\beta}^1$
- Repeat the process for k times until $\hat{\sigma}^{k+1} - \hat{\sigma}^k \leq v$

$$\hat{\sigma}^k := \sqrt{\mathbb{E}_n[Y_i - X_i'\hat{\beta}^k]}$$

Inference in High-Dimensional Linear Regression

FWL Revisited

- How does the predicted value of Y_i change if D_i increases by a unit, *while holding W_i unchanged*?
  - What is the difference in predicted wages between men and women with the same characteristics of human capital?
- In Week 1, we introduced *Frisch-Waugh-Lovell Theorem* (FWL) as a partialling-out method

$$\tilde{Y}_i = \alpha \tilde{D}_i + \tilde{e}_i$$

- where

$$\tilde{D}_i = D_i - \gamma'_{DW} W_i, \quad \tilde{Y}_i = Y_i - \gamma'_{YW} W_i$$
$$\gamma_{DW} = \arg\min_{\gamma} E\left[(D_i - \gamma' W_i)^2\right], \quad \gamma_{YW} = \arg\min_{\gamma} E\left[(Y_i - \gamma' W_i)^2\right]$$

FWL based on Unpenalized OLS Fails in High Dimension

- Not surprisingly, the unpenalized OLS fails in high dimension (p/n is not small) in the *sample* prediction of $\tilde{D}_i$ and $\tilde{Y}_i$
- LASSO can be naturally integrated to reduce dimensionality

Double LASSO Estimation

- Double LASSO satisfies *Neyman Orthogonality*
- Run LASSO regressions of Y_i on W_i and D_i on W_i

$$\hat{\gamma}_{YW} = \arg\min_{\gamma \in \mathbb{R}^p} \sum_i^n (Y_i - \gamma' W_i)^2 + \lambda_1 \sum_j^p |\gamma_j|$$

$$\hat{\gamma}_{DW} = \arg\min_{\gamma \in \mathbb{R}^p} \sum_i^n (D_i - \gamma' W_i)^2 + \lambda_2 \sum_j^p |\gamma_j|$$

$$\check{Y}_i = Y_i - \hat{\gamma}'_{YW} W_i$$

$$\check{D}_i = D_i - \hat{\gamma}'_{DW} W_i$$

Double LASSO Estimation

- Double LASSO satisfies *Neyman Orthogonality*
- In place of LASSO, we can use Post-LASSO or other LASSO relatives
- We run the OLS regression of $\check{Y}_i$ on $\check{D}_i$ to obtain the estimator $\hat{\alpha}$

$$\hat{\alpha} = \arg\min_{\alpha \in \mathbb{R}} \mathbb{E}_n[(\check{Y}_i - \alpha \check{D}_i)] \\ = \frac{\mathbb{E}_n[\check{D}\check{Y}]}{\mathbb{E}_n[\check{D}^2]}$$

- Under *Neyman Orthogonality*, the estimation error in $\check{Y}_i$ and $\check{D}_i$ has no first-order effect on $\hat{\alpha}$

$$\sqrt{n}(\hat{\alpha} - \alpha) \xrightarrow{d} \mathcal{N}(0, V)$$
$$\text{where } V = (E[\tilde{D}_i^2])^{-1} E[\tilde{D}_i^2 e_i^2] (E[\tilde{D}_i^2])^{-1}$$

- With λ_1 and λ_2 found via plug-in method

Double LASSO Estimation

- Good performance of the Double LASSO procedure relies on *approximate sparsity* of the population regression coefficients γ_{YW} and γ_{DW}
- With a sufficiently high speed of decrease in the sorted coefficients and on careful choice of the LASSO parameters
- Absent these guarantees, we cannot theoretically ensure that the first step estimation of $\check{D}_i$ and $\check{Y}_i$ does not have first-order impacts on the final estimator $\hat{\alpha}$
- Practically, LASSO with penalty parameter selected via cross-validation can perform poorly in simulations in moderately sized samples

Invalid Single LASSO Estimation (Naive Method)

- Another intuitive but incorrect LASSO estimator only does LASSO once (*Neyman Ortholonality* not satisfied)
- One applies LASSO regression of Y_i on D_i and W_i to select relevant covariates W_Y, in addition to the covariate of interest, then refits the model using OLS of Y_i on D_i and W_Y

Double LASSO Demonstration in *R*

- See example of testing the Convergence Hypothesis

Inference on Many Coefficients

- Consider the model

$$Y_i = \sum_{\ell=1}^{p_1} \alpha_\ell D_{\ell i} + \sum_{j=1}^{p_2} \beta_j \bar{W}_j + e_i$$

- where we use D_ℓ for $\ell = 1, \ldots, p_1$ to denote the predictors of interest and $\bar{W}_j$ for $j = 1, \ldots, p_2$ to denote other predictors in the model

Inference on Many Coefficients

- There can be at least three motivations for considering many coefficients of interest
  - There can be multiple policies whose effect we would like to infer
  - We can be interested in heterogeneous effects across pre-specified groups
  - We can be interested in nonlinear effects of policies

One-by-One Double LASSO for Many Target Parameters

- Consider the model

$$Y_i = \sum_{\ell=1}^{p_1} \alpha_\ell D_{\ell i} + \sum_{j=1}^{p_2} \beta_j \bar{W}_j + e_i$$

- For each $\ell = 1, \ldots, p_1$, apply the LASSO procedure for estimation and inference on the coefficient α_ℓ in the model

$$Y_i = \alpha_\ell D_{\ell i} + \gamma_\ell' W_\ell + e_i, \quad W_\ell = ((D_k')_{k \neq \ell}, \bar{W}')'$$

where W_ℓ is being partialled out

Other Approaches that Have the Neyman Orthogonality Property

- One way to fix the "single selection" approach is to have *double selection*
- Find controls W_Y that predict Y_i as judged by LASSO
- Find controls W_D that predict D_i as judged by LASSO
- Regress Y_i on D_i and the union of controls $W_Y \cup W_D$
- This procedure is approximately equivalent to the partialling out approach

Other Approaches that Have the Neyman Orthogonality Property

- Another procedure that is approximately equivalent to the partialling out approach is *debiased LASSO*
- Run a LASSO estimator with suitable choice of λ of Y_i on D_i and W_i, and save the coefficient estimate $\hat{\beta}$
- Run a LASSO estimator with suitable choice of λ of D_i on W_i, and save the residualized $\check{D}_i$

$$\hat{\alpha} = \frac{\mathbb{E}_n[(Y_i - W_i'\hat{\beta})\check{D}_i]}{\mathbb{E}_n[D_i\check{D}_i]}$$

- This is similar to the 2SLS estimator—residualized $\check{D}_i$ is used to instrument for D_i ($\check{D}_i \perp W_i$)