Mini-course Machine Learning in Empirical Economic Research

Lecture 3: Penalized regression and applications in treatment evaluation

Andreas Dzemski[1]

[1]University of Gothenburg

June 7, 2019

Setting

Fit a regression curve to model

$$y_i = f(x_i) + \epsilon_i = \beta_0 + x_i'\beta_1 + \epsilon_i$$

- n observations
- $x_i = p_n$-dimensional covariate vector
- $\epsilon_i =$ idiosyncratic error term

Objective

- prediction (for now)
- training error = measure of *in sample* fit

$$\overline{\text{err}}(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{\beta}_0 - x_i'\hat{\beta}_1\right)^2$$

- EPE (expected prediction error) = measure of fit on a new observation

Expected prediction error

- $\mathbb{E}_{\mathcal{T}}$ = expectation operator wrt training sample
- $E_{y,x}$ = integral wrt probability measure of a new (y, x') observation
- assume $\epsilon \perp x$, $\mathbb{E}\epsilon = 0$ and $\mathbb{E}\epsilon^2 = \sigma^2$

$$
\begin{aligned}
EPE(\hat{f}) =& \mathbb{E}_{\mathcal{T}} E_{y,x} \left(y - \hat{f}(x) \right)^2 \\
=& \sigma^2 + E_{y,x} \left\{ \left(f(x) - \mathbb{E}_{\mathcal{T}} \hat{f}(x) \right)^2 + \mathbb{E}_{\mathcal{T}} \left(\hat{f}(x) - \mathbb{E}_{\mathcal{T}} \hat{f}(x) \right)^2 \right\} \\
=& \underbrace{\sigma^2}_{\text{irreducible error}} + \underbrace{E_{y,x} \operatorname{bias}^2 \left(\hat{f}(x) \right)}_{\text{bias}} + \underbrace{E_{y,x} \operatorname{var} \left(\hat{f}(x) \right)}_{\text{variance}}
\end{aligned}
$$

Gauss-Markov assumptions = OLS is BLUE

OLS is unbiased

$$\text{bias}\left(\hat{f}(x)\right) = \mathbb{E}_{\mathcal{T}}\left(\hat{\beta}^{\text{ols}} - \beta\right) x = 0$$

but has potentially large variance

$$\text{var}\left(\hat{f}(x)\right) \approx \frac{\sigma^2 p_n}{n}$$

- OLS is not well-suited for prediction
  - tries to estimate every component β_j
  - doesn't trade off noise and predictive power
- if $p_n \gg n$ then OLS is not even computable

Regression with $p_n \gg n$

- OLS estimator

$$\hat{\beta}^{\text{ols}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}$$

- why is this not computable for $p_n \gg n$?
- Idea of ridge regression:

$$\hat{\beta}^{\text{ridge}} = \left(\mathbf{X}'\mathbf{X} + \lambda\,\text{Diag}\left((0, 1, \ldots 1)'\right)\right)^{-1}\mathbf{X}'\mathbf{y}$$

- $\lambda =$ *regularization* parameter

Ridge regression

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta_0 \in \mathbb{R}, \beta_1 \in \mathbb{R}^{p_n}} \sum_{i=1}^{n} \left(y - \beta_0 - x_i' \beta_1\right)^2 + \lambda \|\beta_1\|_2^2$$

where $\|\beta_1\|_q = \left(\sum_{j=1}^{p_n} |\beta_{1,j}|^q\right)^{1/q}$.

L_q-penalized regression

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta_0 \in \mathbb{R}, \beta_1 \in \mathbb{R}^{pn}} \underbrace{\sum_{i=1}^{n} \left(y_i - \beta_0 - x_i' \beta_1\right)^2}_{\text{loss function}} + \underbrace{\lambda \|\beta_1\|_q^q}_{\text{penalty term}}$$

- because of the penalty term "best" in-sample fit is costly
  - reduces *overfitting*
- cost of choosing "large" coefficients $\Rightarrow$ *shrinkage*
- choice of q = choice of $\lambda \mapsto \mathcal{F}_\lambda$

  q = 2 Ridge regression

  q = 1 Lasso regression (Least absolute shrinkage and selection estimator, Tibshirani 1996)

Intuition of how shrinkage improves prediction

Intercept is not penalized: we will always have (verify!)

$$\hat{\beta}_0^{L_q,\lambda} = \bar{y} - \bar{x}' \hat{\beta}_1^{L_p,\lambda}$$

Predictors of y

$$\mathbb{E}[y] \qquad \text{or} \qquad \mathbb{E}[y \mid x]$$

$$\lambda \to \infty \;\; \|\hat{\beta}_1^{L_p,\lambda} - 0\|_q \to 0 \Rightarrow \text{estimate } \mathbb{E}[y]$$
$$\lambda \to 0 \;\; \|\hat{\beta}_1^{L_p,\lambda} - \hat{\beta}_1^{\text{ols}}\|_q \to 0 \Rightarrow \text{estimate } \mathbb{E}[y \mid x]$$

- L_q-penalized regression "shrinks" towards the unconditional mean
- "shrink" towards a model that is not complex (=unconditional mean)

Choice of λ

- the regularization parameter λ is a *tuning parameter*
- chosen by the empirical researcher
- choose λ to maximize out-of-sample predictive power (we focus on prediction for now)
  - independent validation sample
  - k-fold cross-validation

Simulation study

- all code is on `https://github.com/adzemski/ML_notes`
- sample size $n = 100$
- number of regressors $p_n = 50$

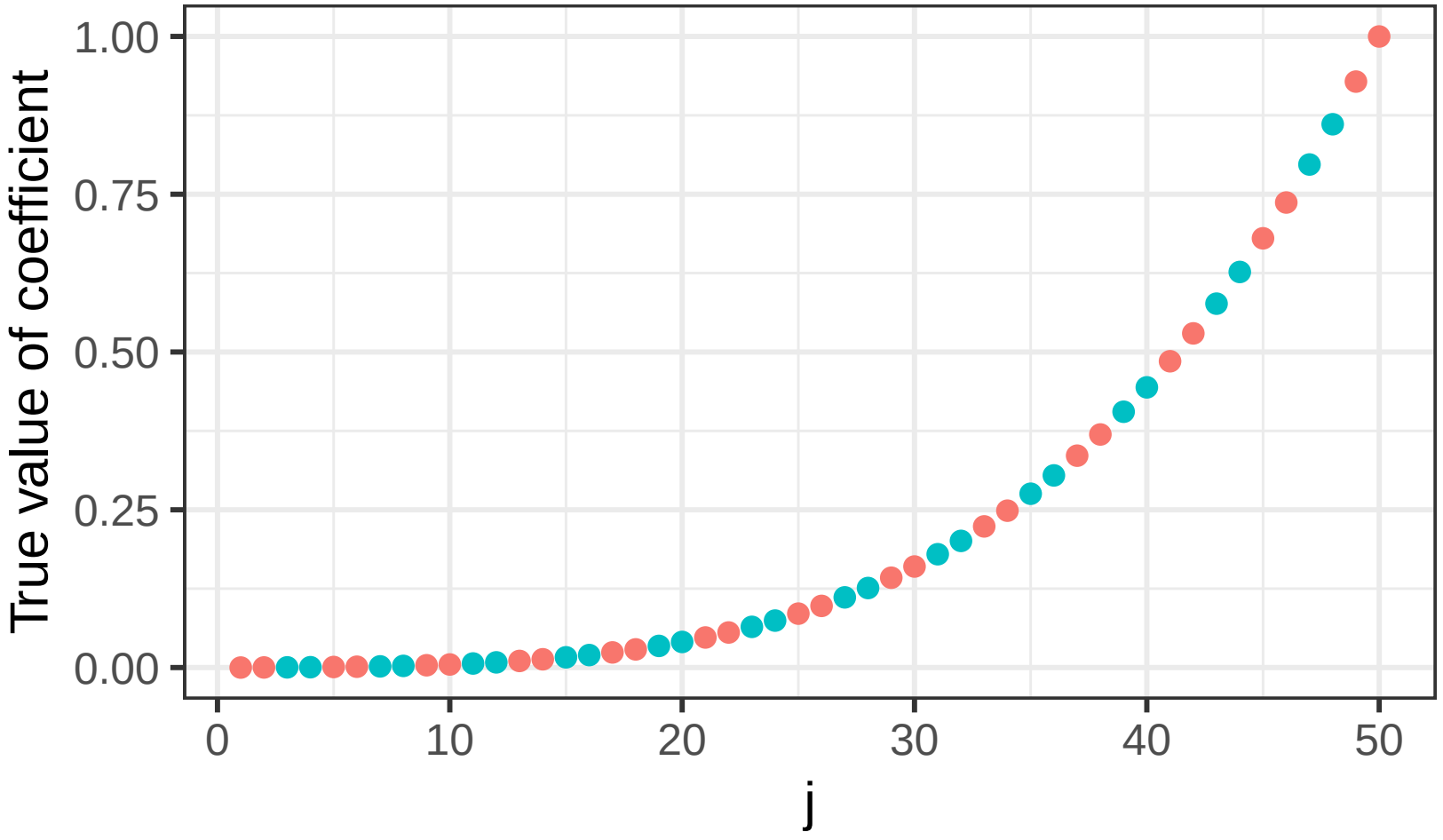

Figure: True values of coefficients

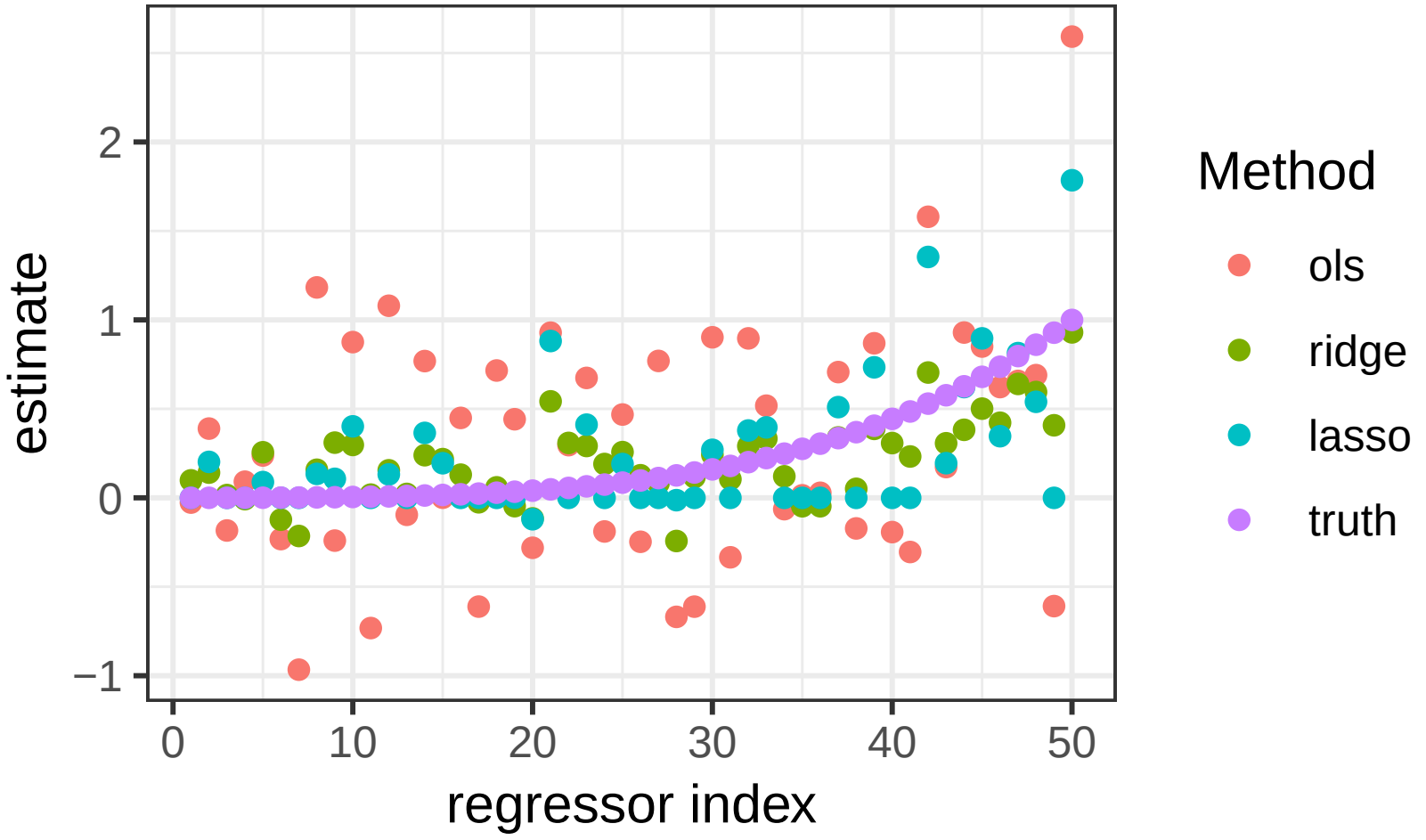

Figure: Estimation results for one sample

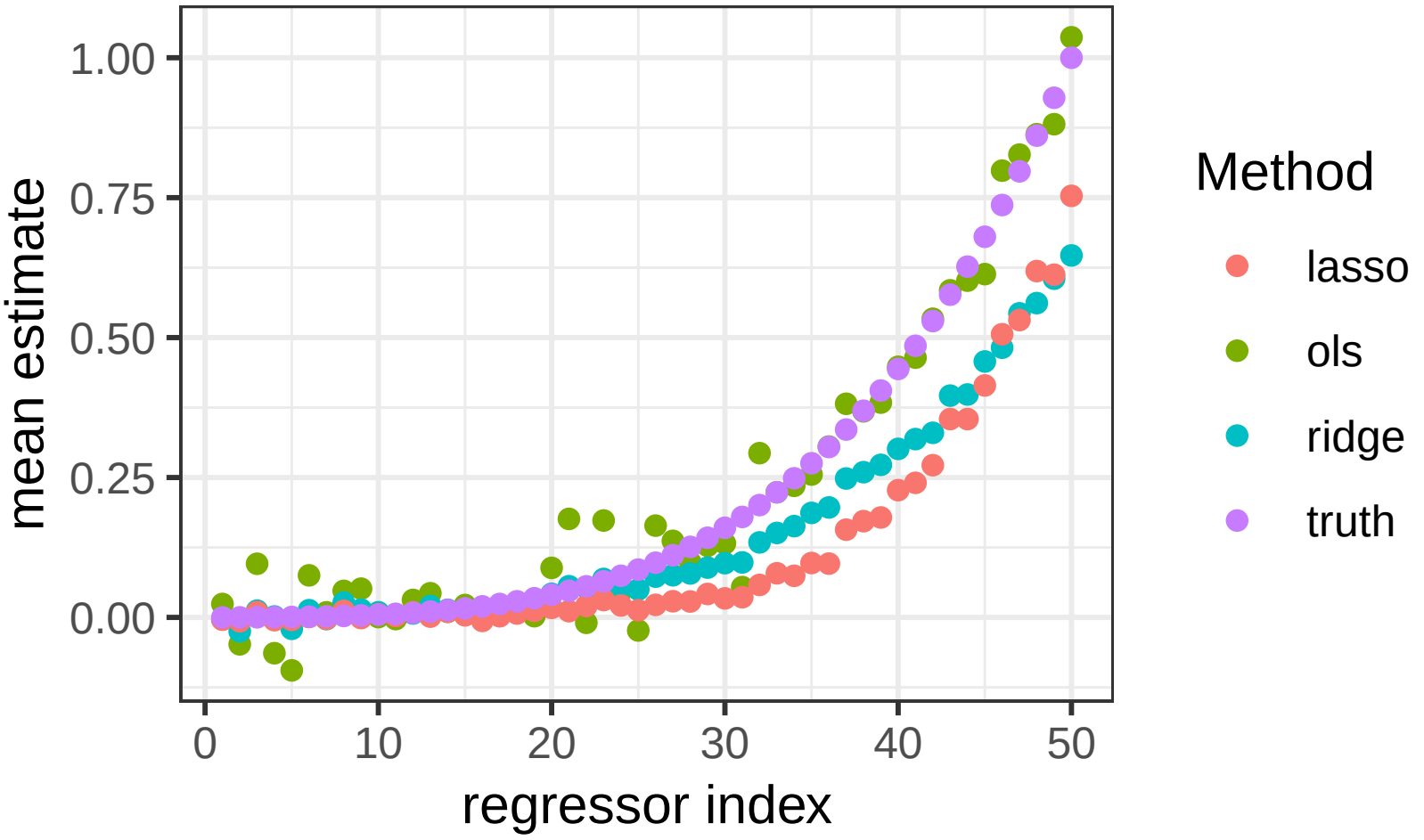

Figure: Expected estimates (average over 200 simulations)

OLS is terrible for prediction

	method	mse
1	ols	27.91
2	ridge	2.89
3	lasso	3.37

Table: Mean-squared-error $MSE(f)$

- Not surprising that Ridge performs best (James-Stein estimator, Empirical Bayes theory)

Variable selection

- an estimator $\hat{\beta}$ selects a variable x_j if $|\hat{\beta}_j| \neq 0$
- variable selection $=$ model selection

Lasso selects variables

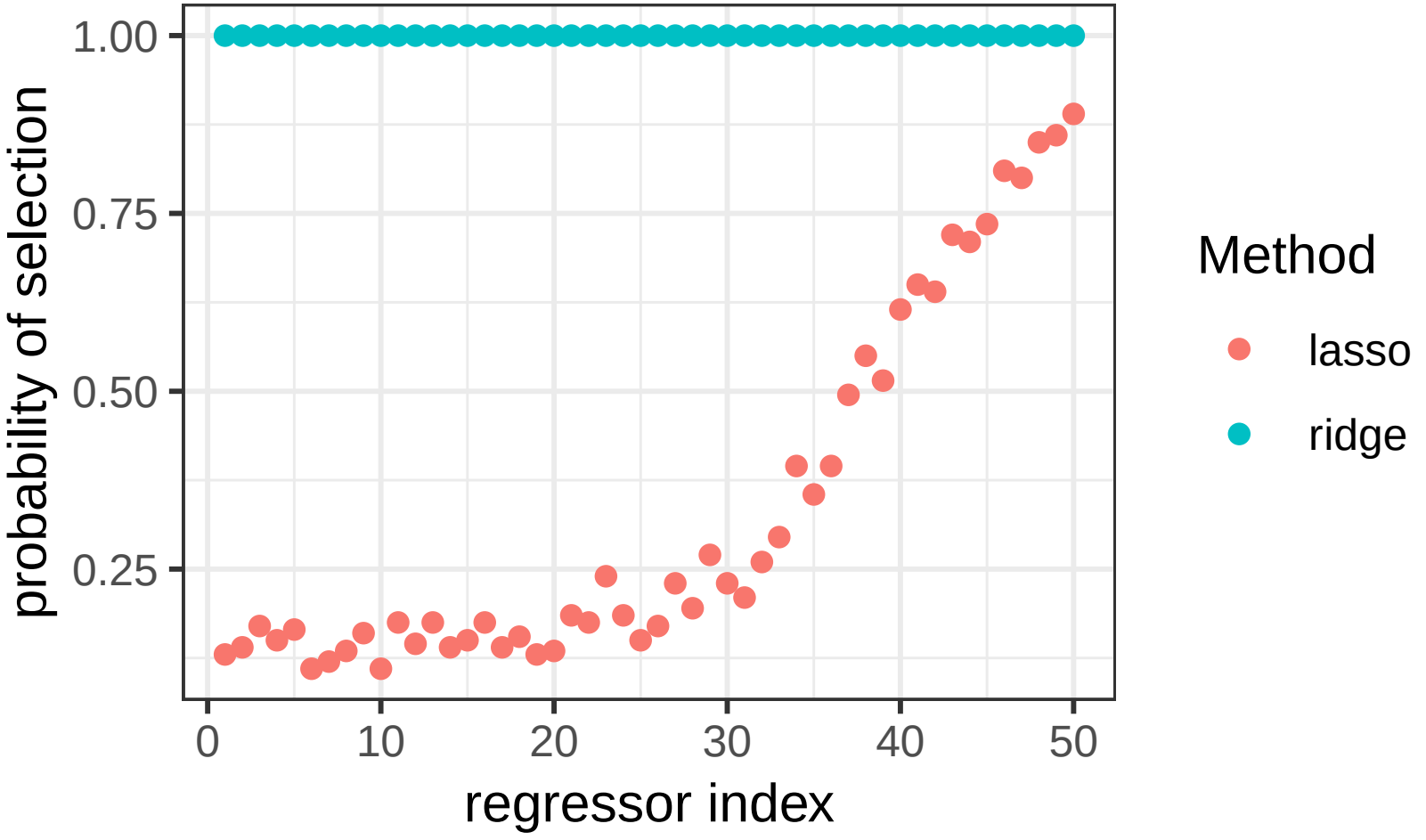

Figure: Probability of including variables (average over 200 simulations)

Instability of variable selection

- is it a problem for prediction?
- for interpretation?

Understanding variable selection by the Lasso

for $p_n = 2$ we solve

$$\min_{\beta} \|\mathbf{y} - \beta_0 - \mathbf{x}_1\beta_1 - \mathbf{x}_2\beta_2\|_2^2$$

$$\text{s.t.} \begin{cases} |\beta_1|^2 + |\beta_2|^2 \leq s & \text{if method} = \text{ridge} \\ |\beta_1| + |\beta_2| \leq s & \text{if method} = \text{lasso} \end{cases}$$

- why?
- recall complexity measure $\mathcal{C}(f)$

Contour sets of the loss function

- $\mathbf{X}_{+1}$ = design matrix including intercept ($n \times (1 + p_n)$)
- $\hat{\beta}^{\text{ols}}$ = OLS estimator including intercept

contour sets

$$\left\{\beta \in \mathbb{R}^{p_n+1} : \|\mathbf{y} - \mathbf{X}_{+1}\beta\|_2^2 = c\right\}$$

are empty or ellipsoids centered at $\hat{\beta}^{\text{ols}}$ (verify!)

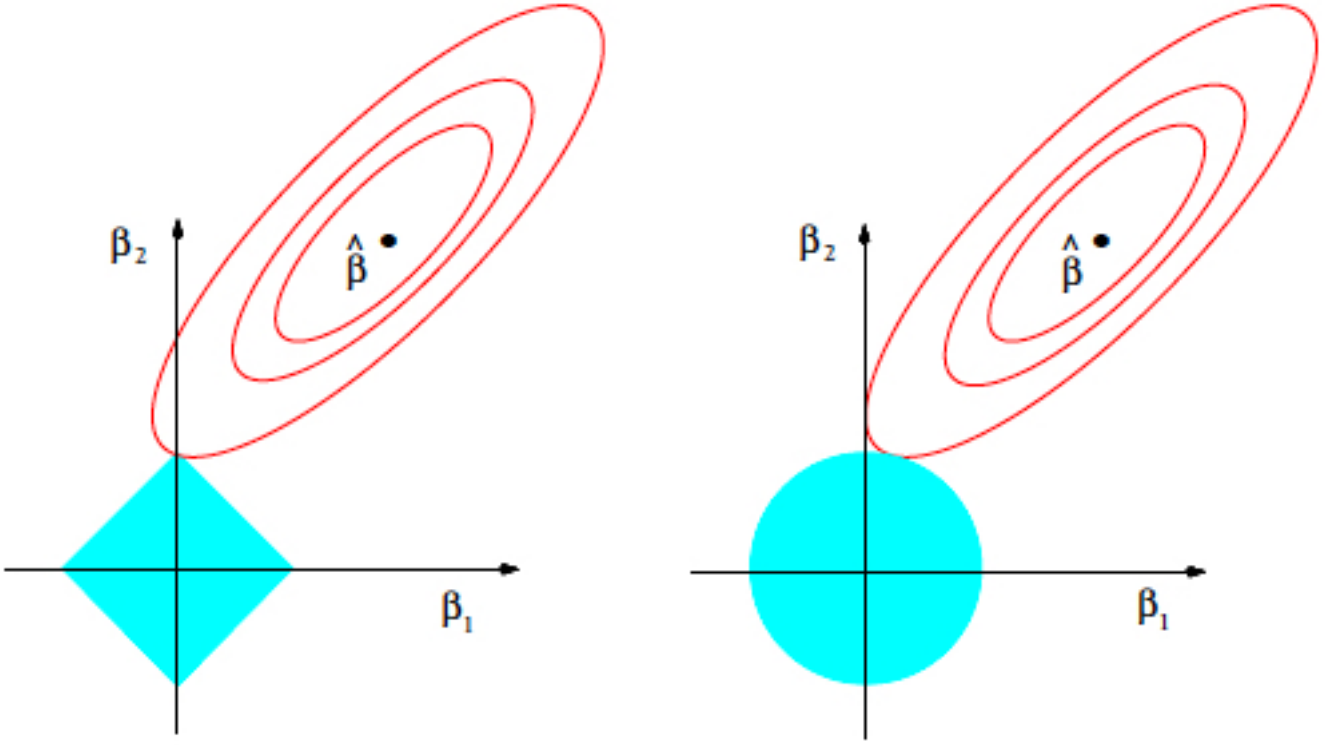

FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions* $|\beta_1| + |\beta_2| \leq t$ *and* $\beta_1^2 + \beta_2^2 \leq t^2$*, respectively, while the red ellipses are the contours of the least squares error function.*

Figure: Figure 3.11 from Hastie, Tibshirani, and J. Friedman (2009)

Series estimation (Newey 1997)

- estimating a smooth regression curve $f : \mathbb{R} \to \mathbb{R}$
- Taylor expansion

$$\begin{aligned} f(x) &= f(0) + f'(0)x + \frac{1}{2}f''(0)x^2 + r(x) \\ &= \underbrace{b_0(x) + b_1(x) + b_3(x)}_{\text{approximation by } p_n = 3 \text{ series terms}} + \underbrace{r(x)}_{\text{“small” remainder}} \end{aligned}$$

- justification as non-parameteric technique
  - $p_n \to \infty$ asymptotics
- may make sense to choose orthogonal basis functions
  - e.g. Legendre polynomials, Fourier basis
- generalization
  - non-smooth functions
  - multi-variate functions

High-dimensional regression vs. non-parametric regression

"We differ from most of the existing literature that considers $p \ll n$ series terms by allowing $p \gg n$ series terms from which we select $s \ll n$ terms to construct the regression fits. Considering an initial broad set of terms allows for more refined approximations of regression functions relative to the usual approach that uses only a few **low-order** *[my emphasis] terms."(Belloni, Chernozhukov, and Hansen 2014)*

A sparse design

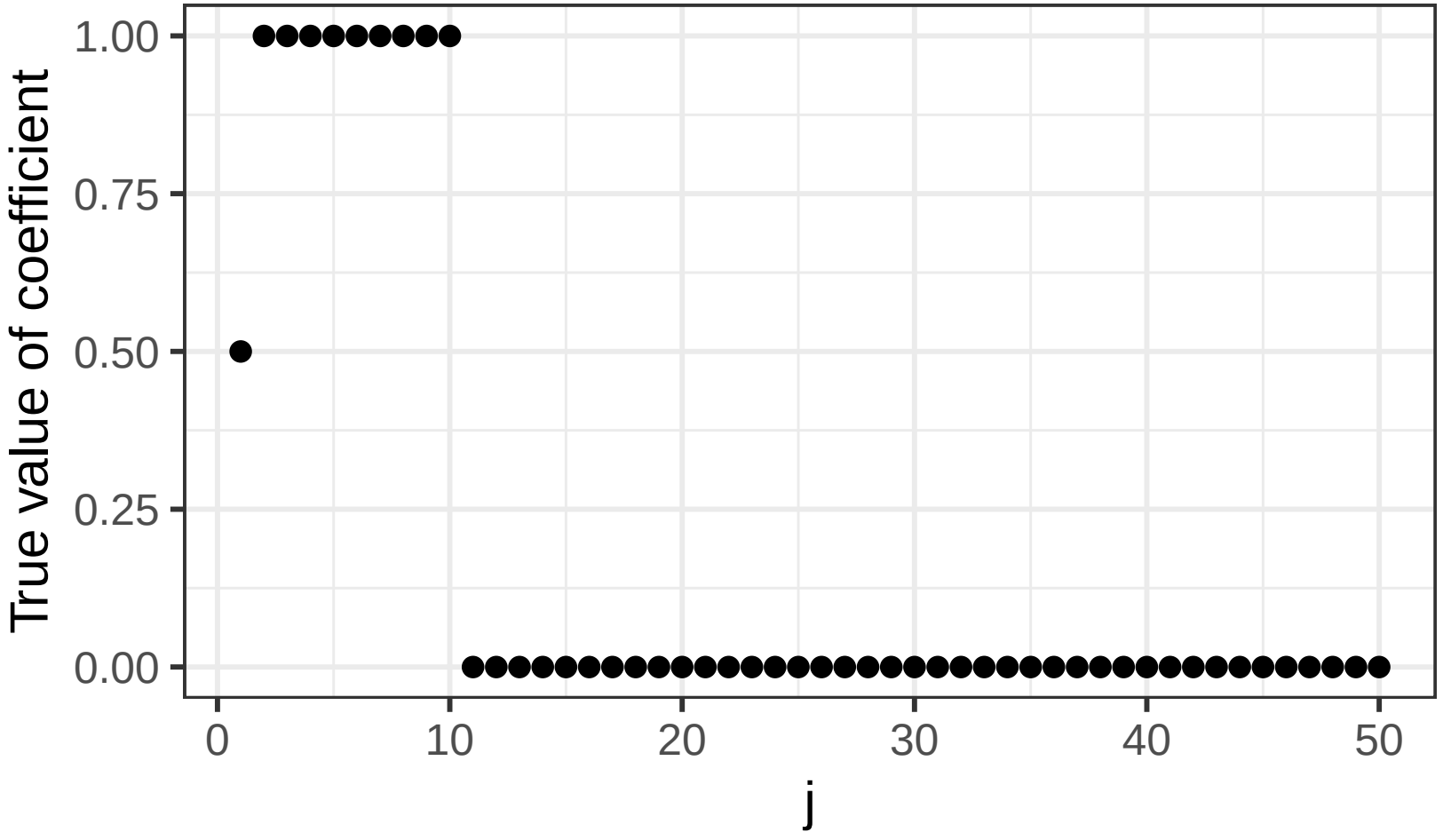

Figure: True values of coefficients

Lasso detects many of the zero coefficients

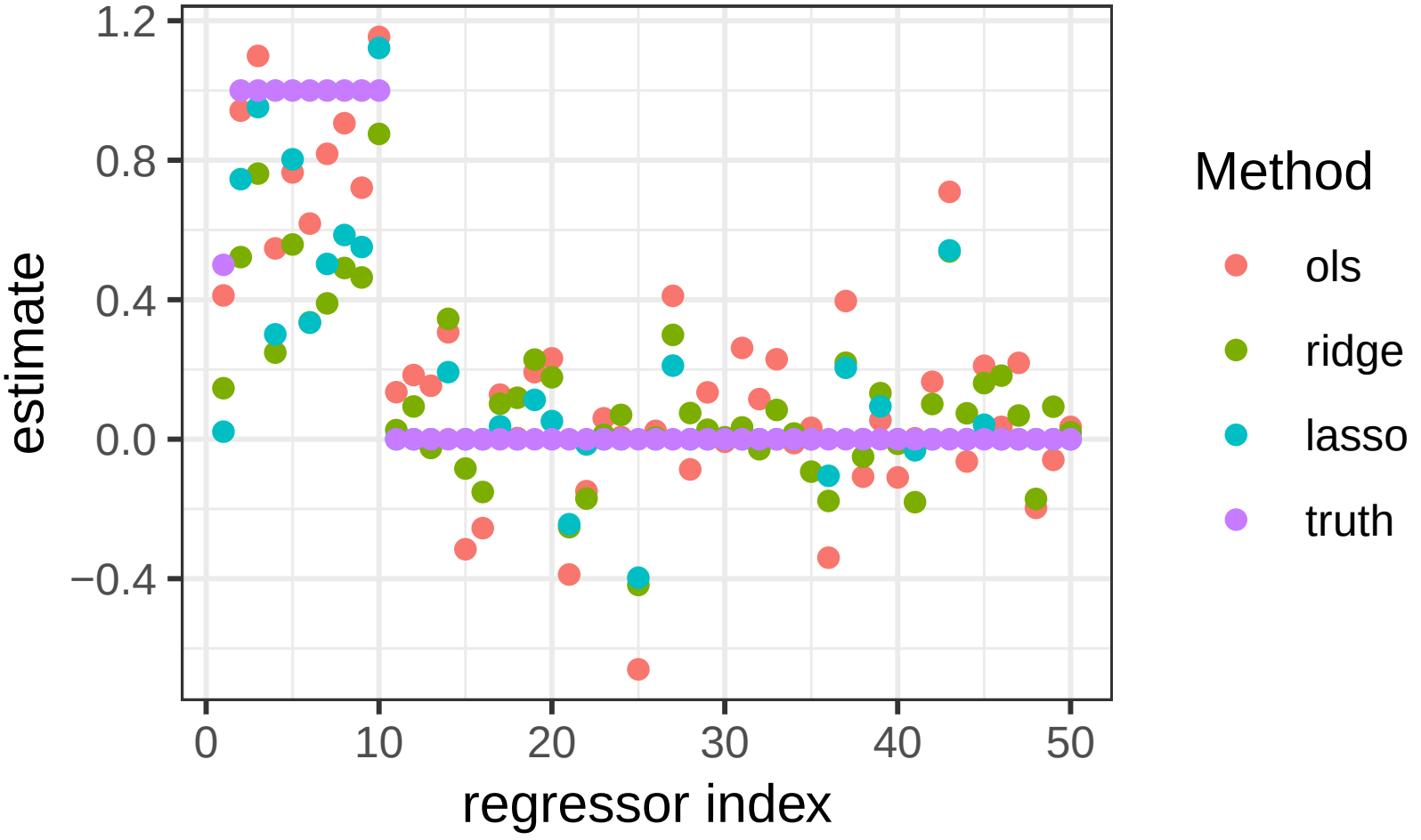

Figure: Estimation results for one sample.

Lasso is good at selecting the true model

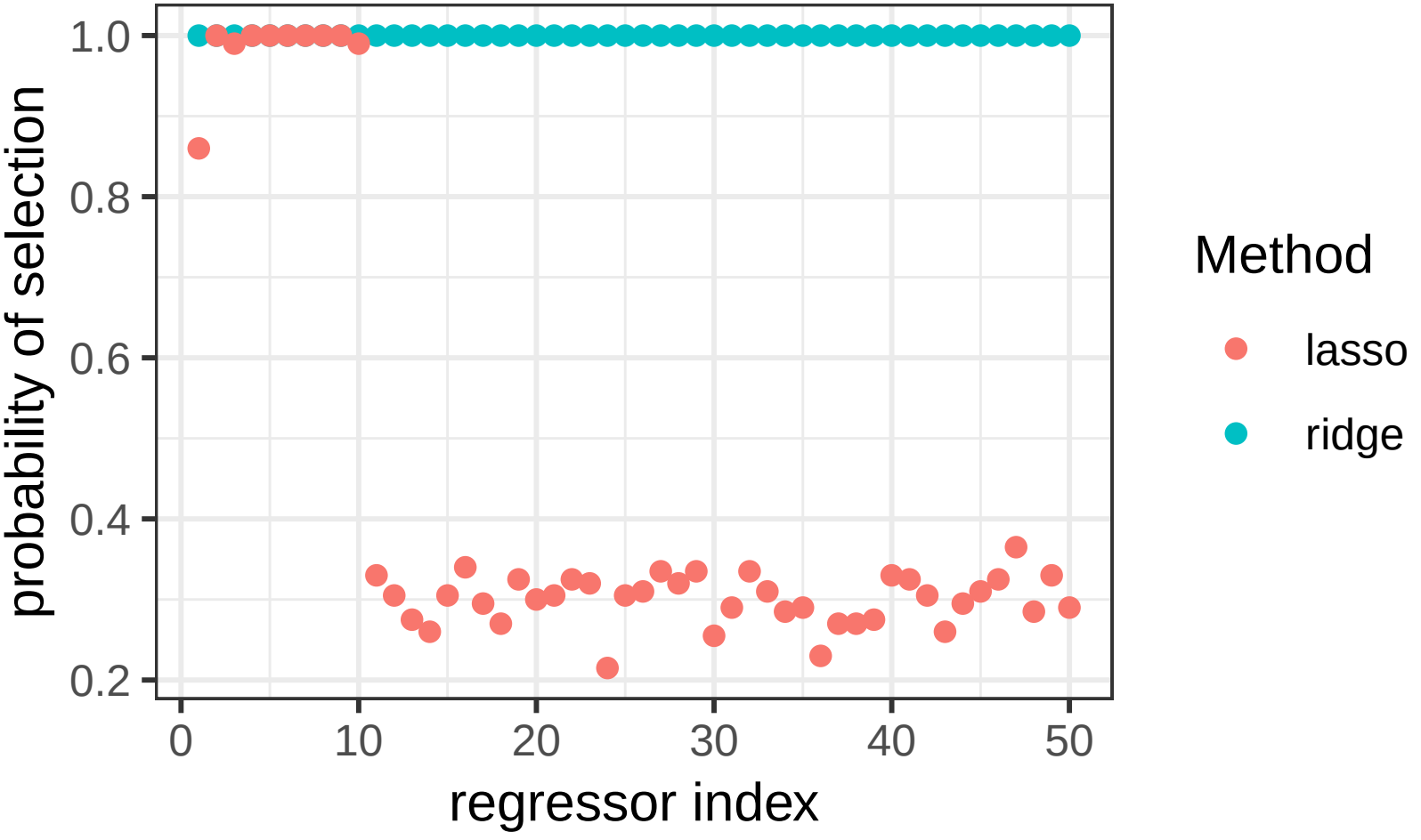

Figure: Probability of including variables (average over 200 simulations).

But still shrinkage

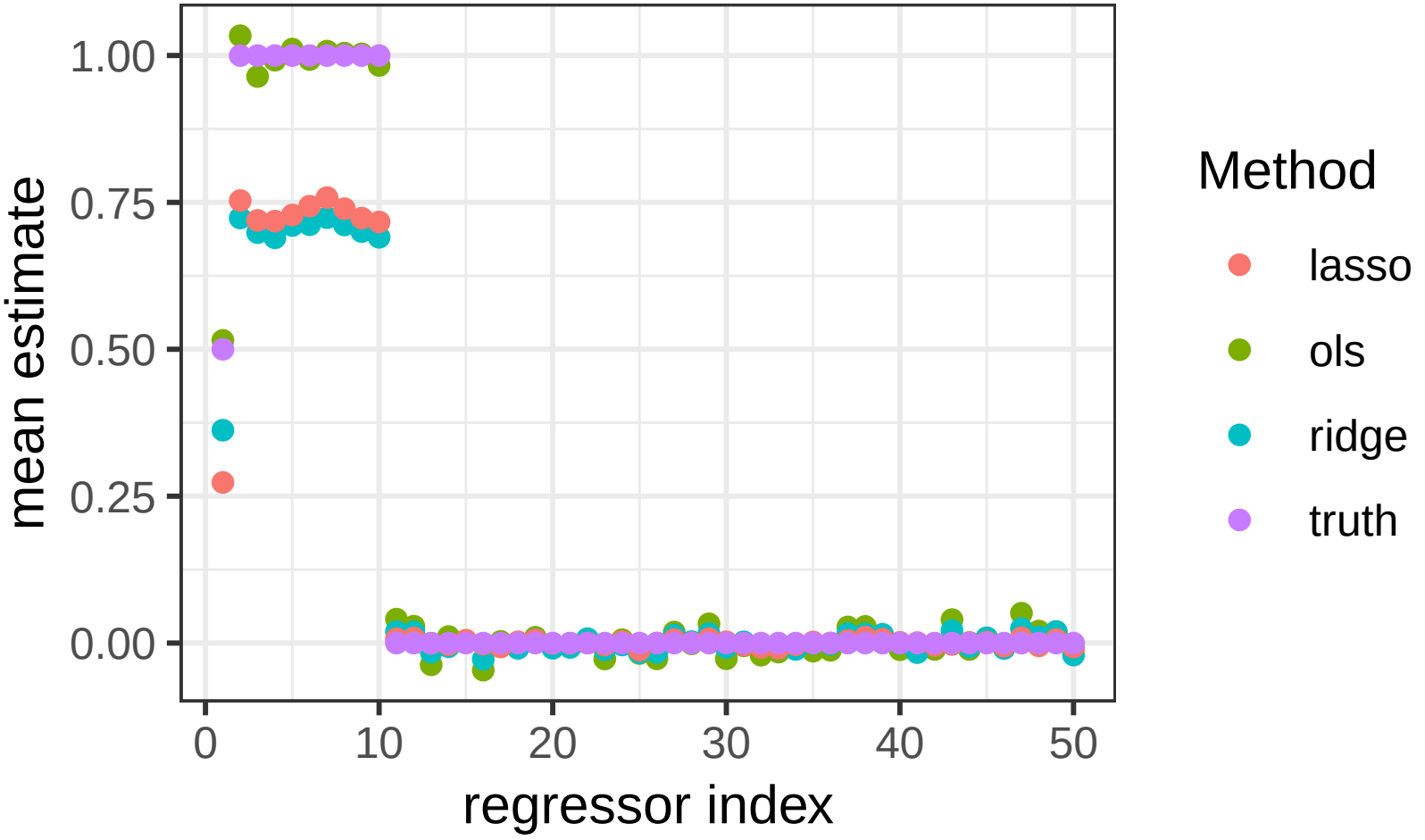

Figure: Expected estimates (average over 200 simulations).

Post-Lasso

- Run lasso on all possible variables
- Select the variable j with $\hat{\beta}_j^{Lasso} \neq 0$
- Run OLS on the selected variable ($=$ post-selection estimator)

Always include a variable

- Post-Lasso makes sense if we are interested in the value of the coefficients
- If there is a specific variable of interest we should always select it
- assume $x_1 =$ treatment dosage
- Lasso solves

$$\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_1 - \sum_{j=2}^{p} x_{j,i} \beta_j \right)^2 + \lambda \sum_{j=2}^{p} |\beta|_j$$

(we don't penalize the intercept and the coefficient on x_1)

Penalty matrix

Lasso solves

$$\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta' x_i\right)^2 + \lambda \|\Psi\beta\|_1$$

where Ψ is a weight matrix. To exclude the first two coefficients from penalization put

$$\Psi = \text{Diag}\left((0, 0, 1, \ldots, 1)'\right).$$

Post-Lasso (uncorrelated design)

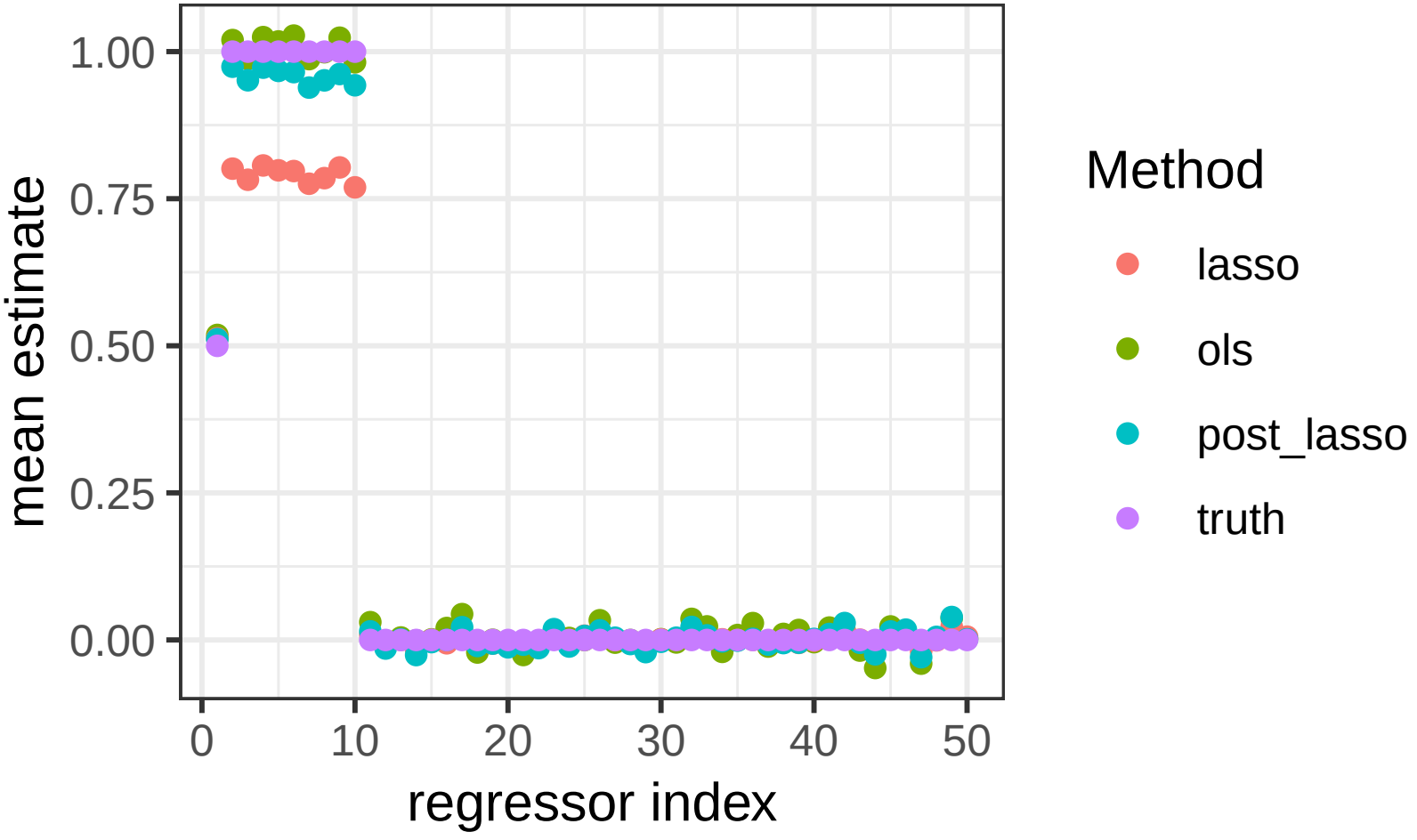

Figure: Expected estimates (average over 200 simulations)

- Why does the post-Lasso *on average* under-estimate the true values?
- Trying to reverse shrinkage has adverse effect on MSE:

	method	mse
1	ols	20.63
2	lasso	1.48
3	post_lasso	20.15

Table: Mean-squared-error $MSE(\hat{f})$

- Is post-lasso estimator $\hat{\beta}_1^{\text{post}}$ a better estimator than $\hat{\beta}_1^{\text{ols}}$? (homework)

Introducing correlation

- In uncorrelated design post selection estimator seems to "work"
- now introduce correlation: $\text{cor}(x_1, x_2) = 0.95$, all other variables uncorrelated

Post-Lasso (correlated design)

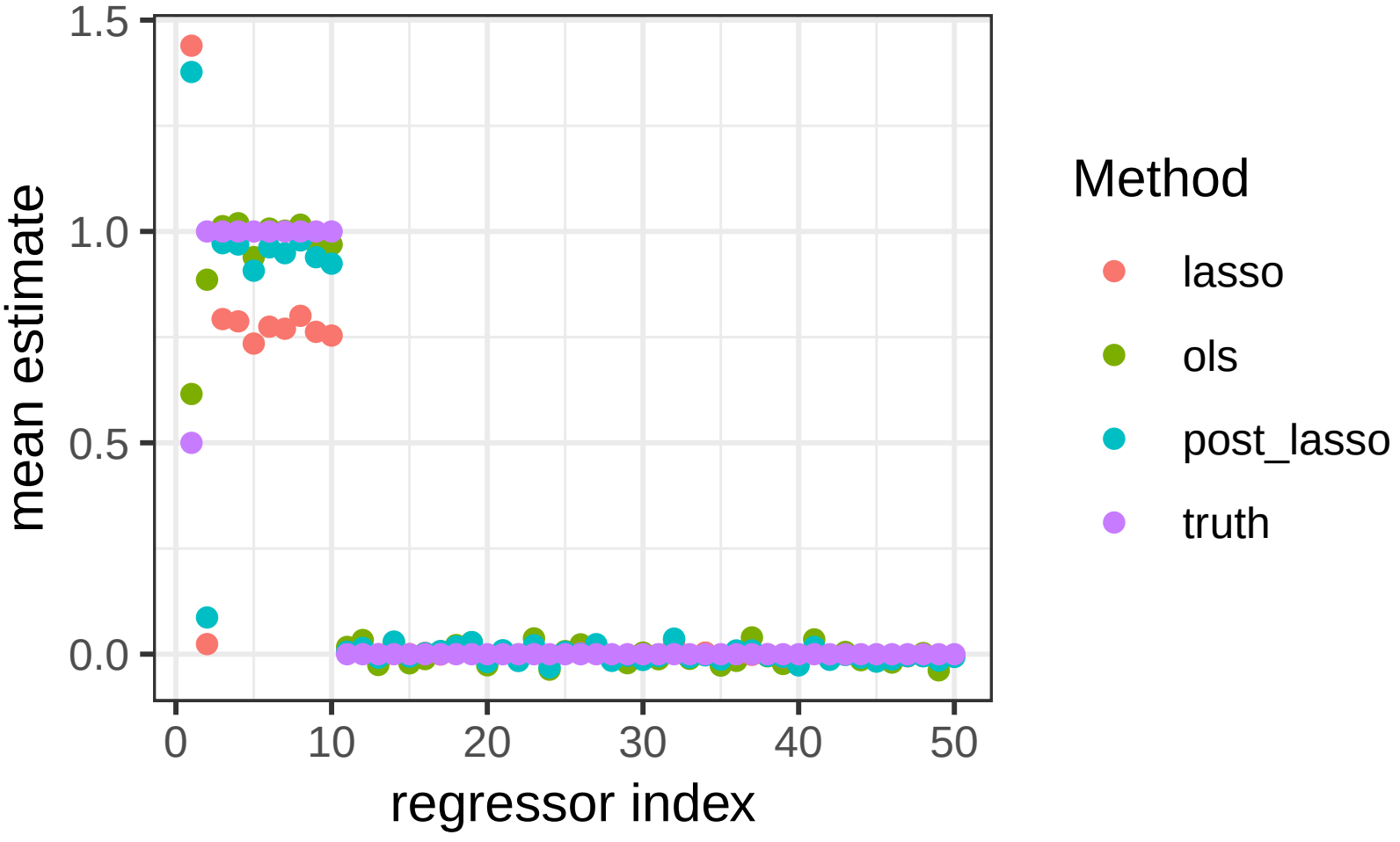

Figure: Expected estimates (average over 200 simulations)

What happened?

- bias of estimated treatment effect = almost 200% of true effect

"Double" selection procedure (Belloni, Chernozhukov, and Hansen 2014)

- intuition: detect variables that are highly correlated with x_1 and make sure they are always selected
- model: based on standard model for "regression type" treatment evaluation model under unconfoundedness

Double selection algorithm

outcome equation:

$$y_i = \alpha_{g,0} + \alpha_1 x_{1,i} + \beta'_{g0} x_{-1,i} + r_{g,i} + \zeta_i$$

selection equation:

$$x_{1,i} = \alpha_{m,0} + \beta'_{m0} x_{-1,i} + r_{m,i} + \nu_i$$

1. variables selected from $x_{-1,i}$ by running Lasso on *outcome* equation $= \hat{I}_1$
2. variables selected from $x_{-1,i}$ by running Lasso on *selection* equation $= \hat{I}_2$
3. run post-Lasso on $\hat{I}_1 \cup \hat{I}_2$

Sparsity assumption

outcome equation:

$$y_i = \alpha_{g,0} + \alpha_1 x_{1,i} + \beta'_{g0} x_{-1,i} + r_{g,i} + \zeta_i$$

selection equation:

$$x_{1,i} = \alpha_{m,0} + \beta'_{m0} x_{-1,i} + r_{m,i} + \nu_i$$

sparsity of linear component:

$$\|\beta_{m0}\|_0 \leq s_n \quad \text{and} \quad \|\beta_{g0}\|_0 \leq s_n,$$

where

$$\|\beta\|_0 = \sum_{j=1}^{p_n} \mathbf{1}(\beta_j \neq 0)$$

Sparsity assumption

size of remainder (= approximation error):

$$\left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}r_{gi}^2\right)^{1/2} \leq \sqrt{\frac{s_n}{n}} \quad \text{and} \quad \left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}r_{mi}^2\right)^{1/2} \leq \sqrt{\frac{s_n}{n}}$$

- s_n has to be small enough

$$\frac{s_n^2 \log^2(n \vee p_n)}{n} = o(1)$$

- Do you expect double selection to fix the "problem" in our simulation above?
- Simulate this yourself (homework).

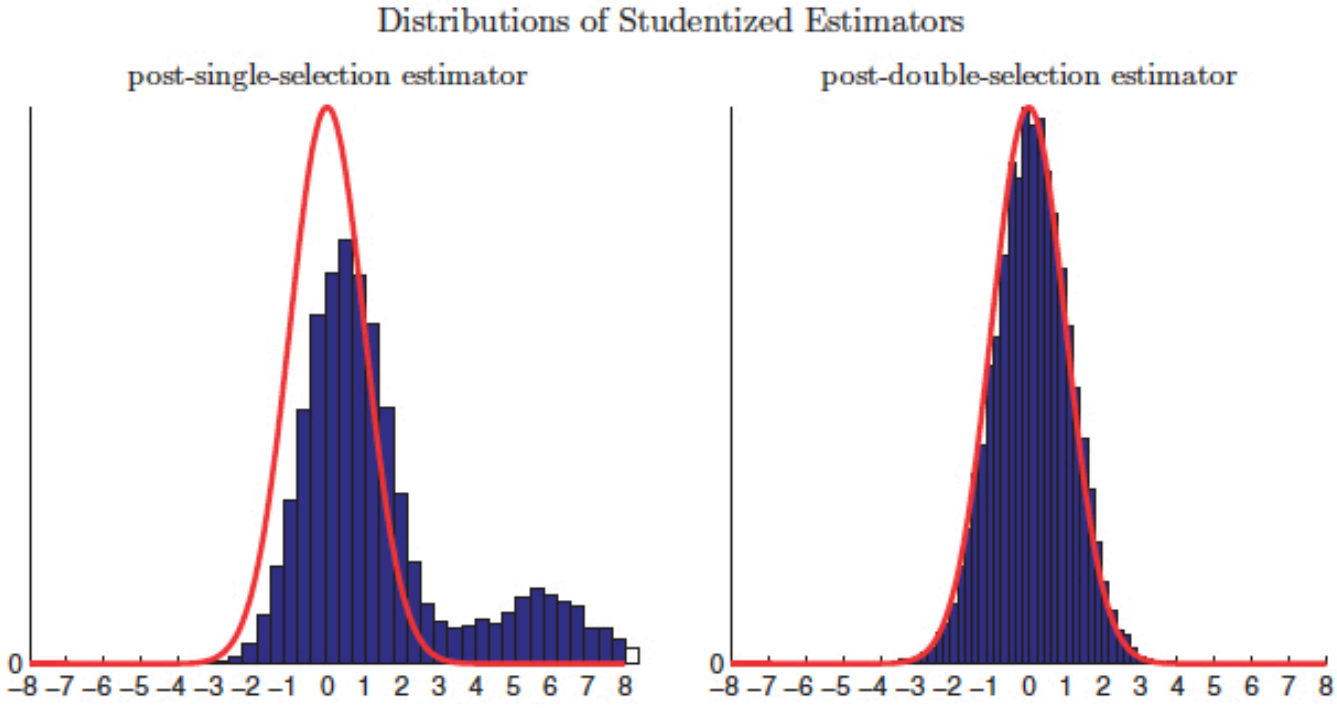

FIGURE 1

The finite-sample distributions (densities) of the standard post-single selection estimator (left panel) and of our proposed post-double selection estimator (right panel). The distributions are given for centered and studentized quantities. The results are based on 10000 replications of Design 1 described in Section 4.2, with R^2's in equation (2.6) and (2.7) set to 0.5.

Figure: Figure 1 from Belloni, Chernozhukov, and Hansen (2014)

Endogenous treatment

- Belloni, Chernozhukov, Fernández-Val, et al. (2017)
- moment condition model
- treatment effect α_1 is defined via moment condition

$$\mathbb{E}_P \psi\left(W, \alpha_1, h_0\right) = 0$$

- h_0 is a functional-valued nuisance parameter that takes values in a space that is approximated well by a "sparse" function space
  - "sparse" = not too complex = entropy is not too large
- the setting in Belloni, Chernozhukov, and Hansen (2014) is a special case (show this!)

More on variable selection through penalization

1. other loss functions
   - example: for binary regression
2. other penalization strategies
3. grouped variables
4. multiple outcomes (multi-target learning)

Selecting variable in ML estimation

- for binary regression L_2-loss not appropriate
- we prefer logit or probit
- generalized linear model

$$\mathbb{E}[y] = g^{-1}(x'\beta),$$

  for *link function* g.
- example: logistic link for logistic regression

$$g(t) = \log\left(\frac{t}{1-t}\right)$$

- maximum likelihood estimation
- loss function = Kullback-Leibler divergence

Ridge and Lasso for generalized linear models

- similar properties to penalized least squares
  - prediction is improved
  - L_1-penalty selects variables
- implemented and ready to use
  - `glmnnet` package for R
- algorithms that can cheaply compute the whole *solution path*
  - solution path $=$ solutions for all λ
  - cross-validiation is cheap

Choice of penalization

- L_q-penalized regression for different q

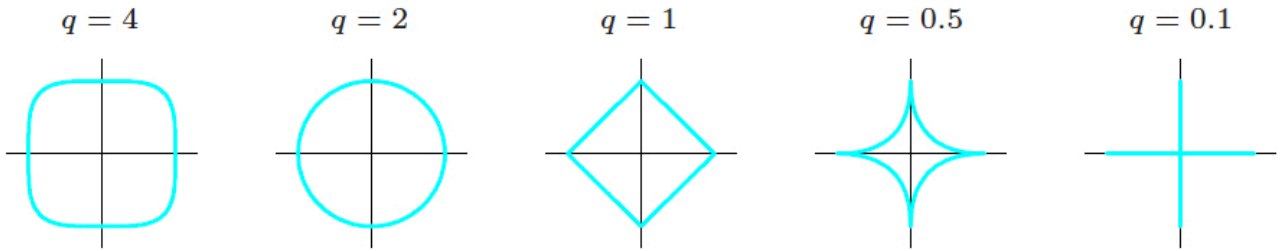

FIGURE 3.12. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for given values of* q.

Figure: Figure 3.12 from Hastie, Tibshirani, and J. Friedman (2009)

Choice of penalization

L_q regression with

$q = 2$ for prediction (Ridge regression)

$q = 1$ for model selection (Lasso).

- middleground?
- we can do L_q-penalized regression $q \in (1, 2)$
- use convex combination of Ridge and Lasso penalties
- *elastic net* penalty

$$\lambda \left(\alpha \|\beta_1\|_2^2 + (1 - \alpha) \|\beta_1\|_1 \right)$$

for $\alpha \in (0, 1)$

L_q penalty vs elastic net

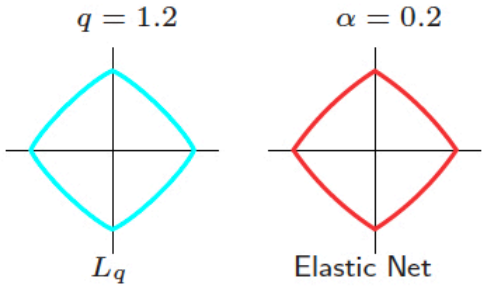

FIGURE 3.13. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for* $q = 1.2$ *(left plot), and the elastic-net penalty* $\sum_j (\alpha\beta_j^2 + (1-\alpha)|\beta_j|)$ *for* $\alpha = 0.2$ *(right plot). Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while the* $q = 1.2$ *penalty does not.*

Figure: Figure 3.13 from Hastie, Tibshirani, and J. Friedman (2009)

- for $q > 1$, L_q penalty is differentiable
- does not select variables
- elastic net selects variables
- cross-validation choice of α?

Penalization based directly on sparsity

Let

$$\|\beta_1\|_0 = \sum_{j=1}^{p_n} \mathbf{1}\{\beta_j \neq 0\}.$$

- sparsity norm
- could try to solve

$$\min_{\beta_0 \in \mathbb{R}, \beta_1 \in \mathbb{R}^{p_n}} \sum_{i=1}^{n} (y_i - \beta_0 - x_i' \beta_1)^2 + \lambda \|\beta_1\|_0$$

- "the L_1-norm [...] provides the tightest convex relaxation of the L_2-norm" (She 2012)
- we shouldn't really have sparsity by itself as a goal

Grouped Lasso

- variables are naturally sorted into L groups
- objective: select groups together
- model

$$y = \beta_0 + \beta_1' x_1 + \cdots + \beta_L' x_L + \epsilon$$

- grouped penalty

$$\lambda \sum_{\ell=1}^{L} w_\ell \|\beta_\ell\|,$$

  for weights $\{w_\ell\}_{\ell=1}^{L}$ and a norm $\|\cdot\|$
- *grouped Lasso* (Bakin et al. 1999; Yuan and Y. Lin 2006)
  - why "Lasso" regardless of choice of $\|\cdot\|$?

Grouped Lasso

- $\|\cdot\| = \|\cdot\|_1$ doesn't work. Why?
- L_q with $q > 1$ works
- usually $q = 2$ so that penalty is

$$\lambda \sum_{\ell=1}^{L} w_\ell \|\beta_\ell\|_2$$

- $\|\beta_\ell\|_2$ is not squared

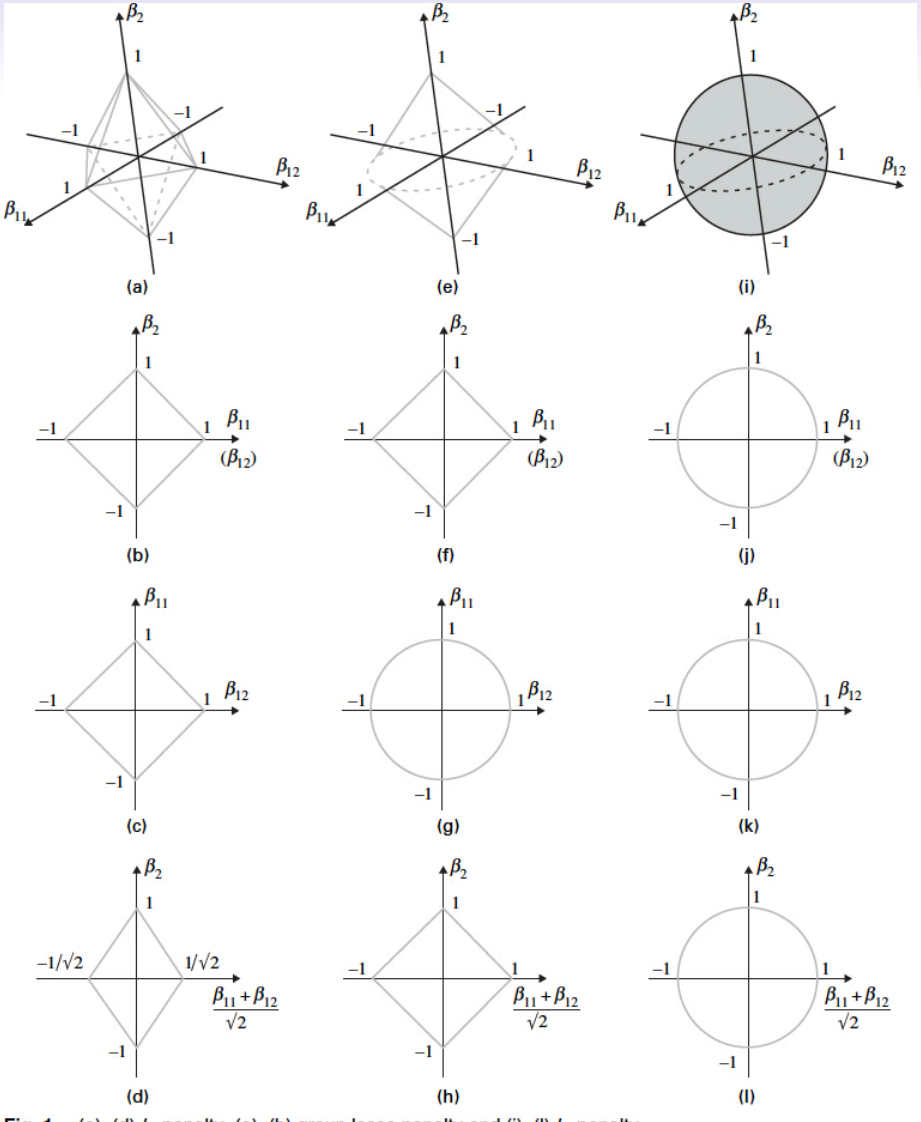

Fig. 1. (a)–(d) l_1-penalty, (e)–(h) group lasso penalty and (i)–(l) l_2-penalty

Figure: Figure 1 from Yuan and Y. Lin (2006): Grouped lasso with $\|\cdot\| = \|\cdot\|_2$ selects variables.

Grouped Lasso

- less common: $\|\cdot\| = \|\cdot\|_\infty$ (Turlach, Venables, and Wright 2005),

$$\lambda \sum_{\ell=1}^{L} w_\ell \max_j |\beta_{\ell,j}|$$

- corresponds to $q = \infty$

Multiple outcomes

- references: Breiman and J. H. Friedman (1997), Turlach, Venables, and Wright (2005), Similä and Tikka (2007), and Sofer, Dicker, and X. Lin (2014)
- loss function for multivariate outcomes
- example with K outcomes:

$$\sum_{k=1}^{K}\sum_{i=1}^{n}\left(y_{i,k}-\beta_{0,k}-\beta_{1,k}'x_i\right)^2=\|\mathbf{Y}-\mathbf{B}_0-\mathbf{X}\mathbf{B}_1\|_F\,,$$

where

$$\mathbf{Y}=\left(y_{i,k}\right)_{\substack{i=1,\ldots,n\\ j=1,\ldots,k}}\quad \mathbf{X}=\left(x_i'\right)_{i=1,\ldots,n}$$

$$\mathbf{B}_0=\left(\beta_{0,k}\right)_{\substack{i=1,\ldots,n\\ k=1,\ldots,K}}\quad \mathbf{B}_1=\left(\beta_{1,k}\right)_{k=1,\ldots,K}$$

- group lasso penalty

Back to treatment evaluation under unconfoundedness

- the approach in Belloni, Chernozhukov, and Hansen (2014) requires sparse model for selection equation
- is *not* robust to misspecification of selection equation (propensity score)

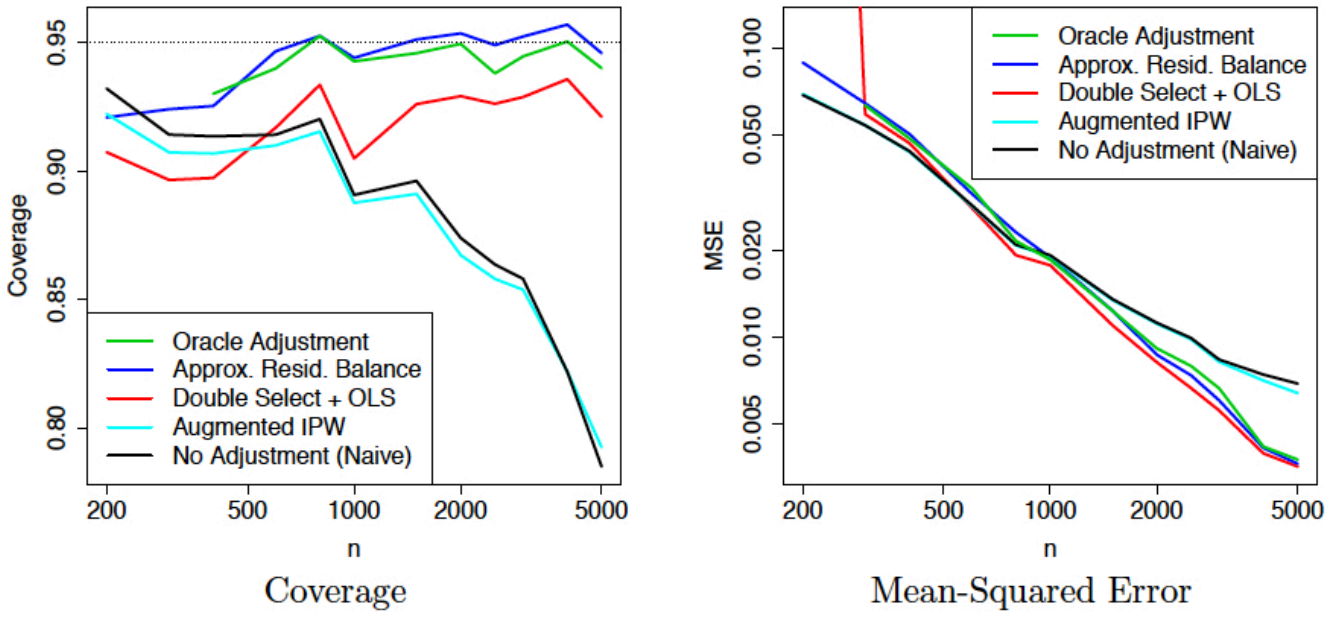

Figure 1: Finite sample performance of the average treatment effect on the treated for different estimators, aggregated over 1,000 replications. The target coverage rate, 0.95, is denoted with a dotted line.

Figure: Figure 1 from Athey, Imbens, and Wager (2018)

Estimating the treatment effect on the treated

- w = binary treatment indicator
- x = control variables (p_n-vector)
- y = outcome variable
- $n_t = \sum_{\{i:w_i=1\}} 1$
- assume unconfoundedness

$$(y(0), y(1)) \perp w \mid x$$

- treatment effect on the treated

$$\tau = \mathbb{E}\left[y(1) - y(0) \mid w = 1\right]$$

Idea of balancing approach

- estimate *control outcome of the treated* by reweighing outcomes in control group

$$\widehat{\mathbb{E}[y(0) \mid w = 1]} = \sum_{i:w_i=0} \hat{\gamma}_i y_i$$

- weight sequence $\{\hat{\gamma}\}_{i:w_i=0}$ balances covariates in treatment and control group

$$\frac{1}{n_t} \sum_{\{i:w_i=1\}} x_{i,j} \approx \sum_{\{i:w_i=0\}} \hat{\gamma}_i x_{i,j} \qquad \text{for } j = 1, \ldots, p_n.$$

Robustness of balancing approach

- *balancing methods* are often robust to misspecification of propensity score (J. M. Robins and Ritov 1997)
- examples of rebalancing methods
  - inverse probability weighting (Cassel, Särndal, and Wretman 1976; J. Robins 1986)
  - inverse probability tilting (Graham, Pinto, and Egel 2012)

Athey, Imbens, and Wager (2018)

- combine regression and rebalancing techniques
- outcome model

$$y_i = \begin{cases} \beta_c' x_i + \epsilon_i & \text{if } w_i = 0 \\ w_i y_i & \text{if } w_i = 1 \end{cases}$$

- β_c is sparse

$$\|\beta_c\|_0 \leq s_n \quad \text{and} \quad \frac{s_n \log(p_n)}{n} = o(1)$$

A regression approach

- $n_t = \sum_{\{i:w_i=1\}} 1$
- $n_c = \sum_{\{i:w_i=0\}} 1$
- $\bar{x}_t = \sum_{\{i:w_i=1\}} x_i / n_t$
- $\bar{y}_t = \sum_{\{i:w_i=1\}} y_i / n_t$

$$\hat{\tau} = \bar{y}_t - \bar{x}_t' \hat{\beta}_c$$

- Belloni, Chernozhukov, and Hansen (2014) show that $\hat{\beta}_c =$ post-single-selection estimator does not work well (OV bias)

A balancing approach

balancing estimator

$$\hat{\tau} = \bar{y}_t - \sum_{\{i:w_i=0\}} \hat{\gamma}_i y_i$$

- $\{\hat{\gamma}_i\}_{i:w_i=0}$ weight sequence that re-weighs covariates in control group so that the covariate distribution in control group "looks like" covariate distribution in treatment group

$$\sum_{i:w_i=0} \hat{\gamma}_i = 1$$

Choice of weights

- $\hat{e}(x_i) =$ estimator of propensity score
- inverse probability weighting

$$\hat{\gamma}_i \propto \frac{\hat{e}(x_i)}{1 - \hat{e}(x_i)}$$

- does not enforce exact balance
- in finite dimensions: does not achieve the semi-parametric efficiency bound (Graham, Pinto, and Egel 2012)

Intuition for robustness of re-balancing

"...in a linear model, the bias for estimators based on weighted averaging depends solely on $\bar{x}_t - \sum_{\{i:w_i=0\}} \hat{\gamma}_i x_i$. *Therefore, getting the propensity model exactly right is less important than accurately matching the moments of* $\bar{x}_t$. *In high dimensions, however, exact re-balancing weights do not in general exist." (Athey, Imbens, and Wager 2018)*

Approximate residual balancing

$$\hat{\tau} = \bar{y}_t - \bar{x}_t \hat{\beta}_c - \overbrace{\sum_{\{i:w_i=0\}} \hat{\gamma}_i \underbrace{\left(y_i - x_i \hat{\beta}_c\right)}_{\text{average over this} = \text{bias in control group}}}^{\text{approximate OV bias in treatment group}}$$

- $\{\hat{\gamma}_i\}_{i:w_i=0}$ "approximately" balances treatment and control group

Estimator $\hat{\beta}_c$

- Lasso estimator

$$\hat{\beta}_c = \arg\min_\beta \left\{ \sum_{i:w_i=0} (y_i - x_i'\beta)^2 + \lambda \|\beta\|_1 \right\}$$

- selection is based *only on the outcome equation*
- Lasso estimator, not post-Lasso (why?)
- tuning parameter λ

Weight estimation

- $\mathbf{X}_c$ = design matrix in control group $(n \times p_n)$

weight estimation

$$\hat{\gamma} = \arg \min_{\gamma \in \mathbb{R}^{n_c}} \left\{ \zeta \max_{j=1,\ldots,p_n} (\bar{x}_t - \mathbf{X}_c' \gamma)_j^2 + (1-\zeta)\|\gamma\|_2^2 \right.$$
$$\left. \text{s.t.} \sum_{i:w_i=0} \gamma_i = 1 \text{ and } 0 \leq \gamma_i \leq n_c^{-2/3} \right\}$$

- tuning parameter ζ

Simulation results for root-mean-squared error

Beta Model	dense		harmonic		moderately sparse		very sparse	
Propensity Model	dense	sparse	dense	sparse	dense	sparse	dense	sparse
Naive	6.625	7.119	3.557	3.924	1.257	1.256	0.711	0.722
Elastic Net	4.328	1.058	2.190	0.665	0.716	0.350	0.237	0.204
Approximate Balance	**3.960**	1.179	2.130	0.686	0.789	0.362	0.464	0.316
Approx. Residual Balance	**3.832**	**0.423**	**1.854**	**0.320**	**0.495**	**0.213**	**0.185**	**0.165**
Inv. Propensity Weight	5.341	3.094	2.866	1.707	1.026	0.596	0.586	0.398
Augmented IPW	4.082	0.618	2.031	0.415	0.607	0.242	0.209	**0.166**
Weighted Elastic Net	4.086	0.562	1.984	0.385	0.575	0.232	0.207	**0.171**
TMLE Elastic Net	**3.811**	0.591	**1.843**	0.399	**0.495**	0.239	**0.192**	**0.165**
Double-Select + OLS	6.625	0.620	3.540	0.430	0.525	0.233	0.254	**0.165**

Table 1: Root-mean-squared error $\sqrt{\mathbb{E}\left[(\hat{\tau}-\tau)^2\right]}$ in the two-cluster setting. We used $n = 500$, $p = 2000$, and scaled the signal such that $\|\beta\|_2 = 2$. All numbers are averaged over 400 simulation replications.

Figure: Table 1 from Athey, Imbens, and Wager (2018).

Wrap up of Part I

- supervised learning = statistical methods for prediction
- causal inference includes predictive tasks
- two-step process
  - choose hyperparameters
  - train the model
- prediction tasks can be validated
  - validation data set
  - cross-validation
- linear model: coefficient interpretation
  - for predictive tasks coefficient may have no economic interpretation
  - shrinkage
  - post-lasso: omitted variable bias