

Regularised methods

Oxford Spring School in Advanced Research Methods 2025

Dr Thomas Robinson, LSE

Day 2/5

Introduction

Yesterday we explored how a familiar estimator (logistic regression) incorporates some fundamental aspects of ML

- ▶ ML is not some entirely new, alien type of doing statistics
- ▶ ML is typically focused on prediction problems
- ▶ Lots of really useful ML models are extensions of regression framework

So how should we understand OLS within a prediction context?

How do other popular forms of ML come out of it?

Today's session

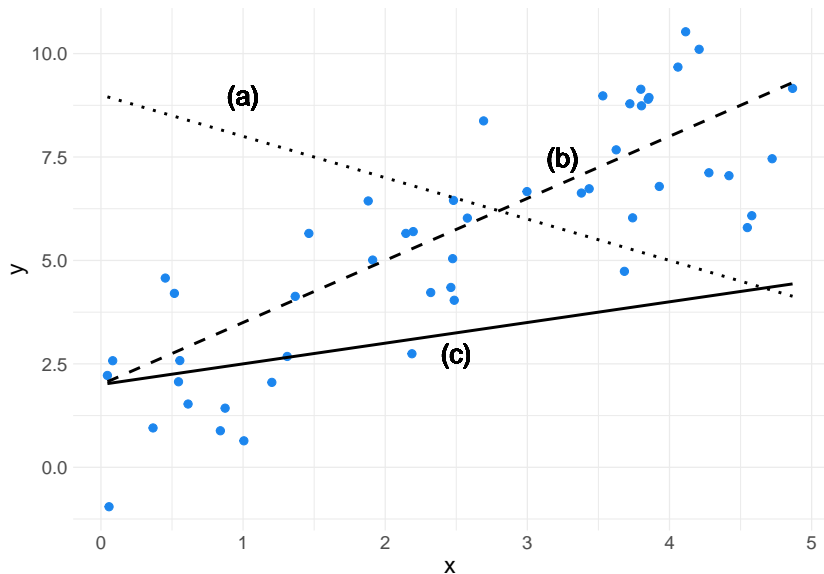
1. Recap OLS from prediction perspective
 - ▶ How does OLS work?
 - ▶ Optimisation criteria
 - ▶ Bias-variance trade-off
2. LASSO estimator
3. Practical application

Key topics:

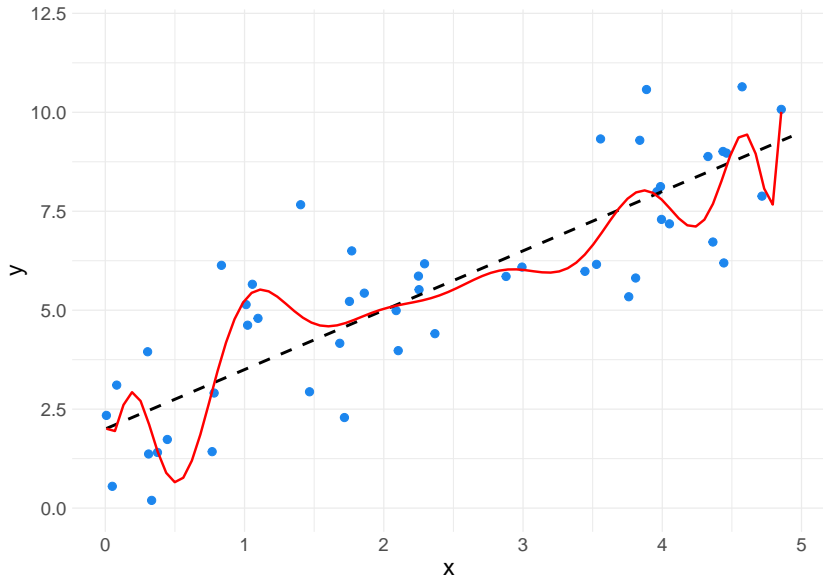
- ▶ Bias and variance
- ▶ Regularisation

Ordinary Least Squares Regression

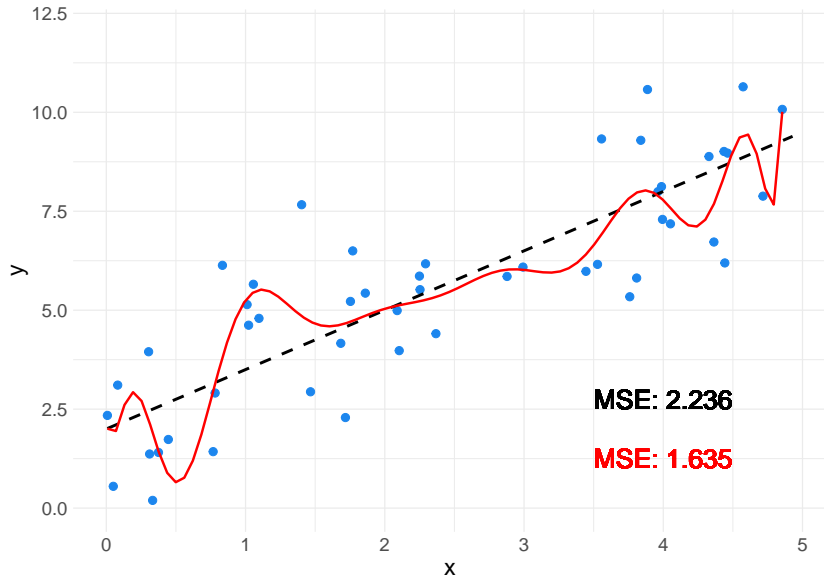
Refresher: which is the best line?



Refresher: which is the best line? 2



Refresher: which is the best line? 2



OLS as a tool for inference

In inference terms, OLS estimates $\hat{\beta}$ that:

- ▶ Capture the linear relationships between \mathbf{X} and \mathbf{y}
- ▶ Yield individual estimates of the partial associations between \mathbf{X} on \mathbf{y}
- ▶ Allows us to understand the uncertainty over $\hat{\beta}$
 - ▶ E.g., how confident we are that there is a non-zero relationship between x_1 and \mathbf{y}

OLS: Optimisation

OLS regression minimises the **sum of the squared error** between the regression line ($\mathbf{X}\beta$) and the observed outcome (\mathbf{y}):

$$\arg \min_{\beta} \sum_{i=1}^N (y_i - \mathbf{x}_i \beta)^2,$$

where $\mathbf{x}_i \beta$ is the linear regression function.

How might we solve this?

- ▶ Maximum likelihood estimation
- ▶ Calculus – there is a closed form solution (unlike logistic regression)

Why do we like OLS?

Not only does OLS have a closed form solution, we also know that, under the Gauss Markov (GM) assumptions, OLS is:

- ▶ **B**est
- ▶ **L**inear (Hansen 2021)
- ▶ **U**nbiased
- ▶ **E**stimator

GM Assumptions

In other words, in terms of estimating the parameters β , you will not find a model with a lower variance that is also unbiased

OLS is good for inference:

- ▶ Typically very concerned with generating **unbiased** estimates of $\hat{\beta}$
- ▶ Most efficient test of a (linear) hypothesis

Bias

Bias is a feature of the estimator:

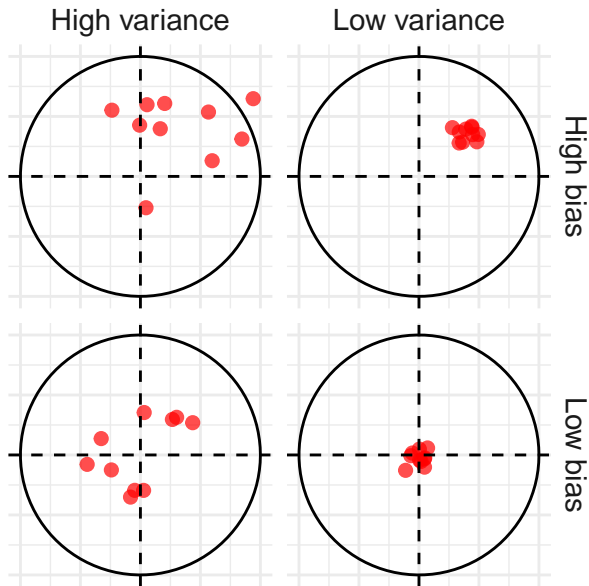
- ▶ $\text{Bias}_\beta = (\mathbb{E}[\hat{\beta}] - \beta)$
- ▶ On average, the estimated parameters are equal to the true parameters
- ▶ Under GM, we know that $(\mathbb{E}[\hat{\beta}] - \beta) = 0$

Variance

As we sample new (GM-satisfying) data, the parameters of our model will shift:

- ▶ Hence, there will be variance over our parameter estimates
- ▶ $V_{\hat{\beta}} = \mathbb{E}[(\mathbb{E}[\hat{\beta}] - \hat{\beta})^2]$
- ▶ The average distance between a particular parameter estimate and the mean of parameter estimates over multiple samples

Visualising bias and variance



Predicting *new* values

In the remainder of today's session, we are going to consider the following generic supervised learning problem:

- ▶ We observe $(\mathbf{y}, \mathbf{X}) \in \mathcal{D}$
 - ▶ A training sample that is taken from a wider possible set of data
 - ▶ I.e. we can think, counterfactually, of resampling to get a new sample $(\mathbf{y}_{\text{New}}, \mathbf{X}_{\text{New}})$
- ▶ We also observe a “test” dataset \mathbf{X}'

The goal is to estimate \mathbf{y}' by training a model \hat{f}

- ▶ The outcomes that correspond to \mathbf{X}'

OLS as a tool for prediction

When we run OLS, we also get a “trained model”:

- ▶ \hat{f} – that has parameters equal to $\hat{\beta}$
- ▶ Can be applied to a new “test” dataset \mathbf{X}'
- ▶ To generate new predictions \mathbf{y}'

Bias and variance of predictions

We can also think of bias in terms of the predictions:

- ▶ $\text{Bias}_y = (\mathbb{E}[\hat{y}] - y)$
- ▶ We ideally want low bias
- ▶ High bias suggests the model is not sensitive enough

And we can think about the variance of the prediction:

- ▶ $\mathbb{V}_{\hat{y}} = \mathbb{E}[(\mathbb{E}[\hat{y}] - \hat{y})^2]$

High variance means that the model is very sensitive to \mathbf{X} – the training data – but will perform poorly out-of-sample

- ▶ With new data, and high variance, we would expect quite different predictions

Bias-variance trade off

So can't we just choose a low-variance, low-bias modeling strategy?

Assume we could calculate the mean squared error of some test data \mathbf{X}' given a trained model \hat{f} :

$$\text{MSE} = \mathbb{E}[(\hat{f}(\mathbf{X}') - \mathbf{y}')^2].$$

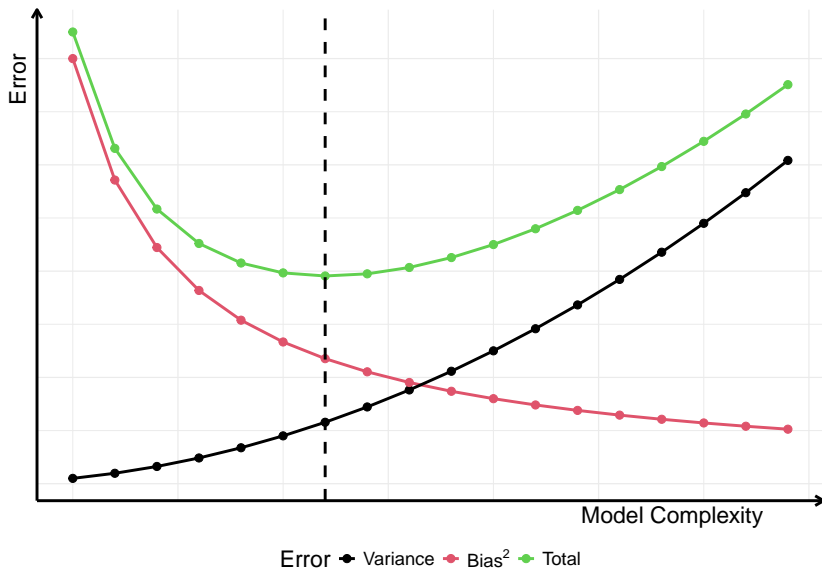
We can decompose this further:

$$MSE = \underbrace{\mathbb{E}[(\hat{f}(\mathbf{X}') - \mathbb{E}[\hat{\mathbf{y}}])^2]}_{\text{Variance}} + \underbrace{(\mathbb{E}[\hat{\mathbf{y}}] - \mathbf{y}')^2}_{\text{Bias}^2}$$

So holding the MSE fixed, if we reduce the variance we must increase the bias

- I.e. there is a **bias-variance trade-off**

Visualising the trade-off

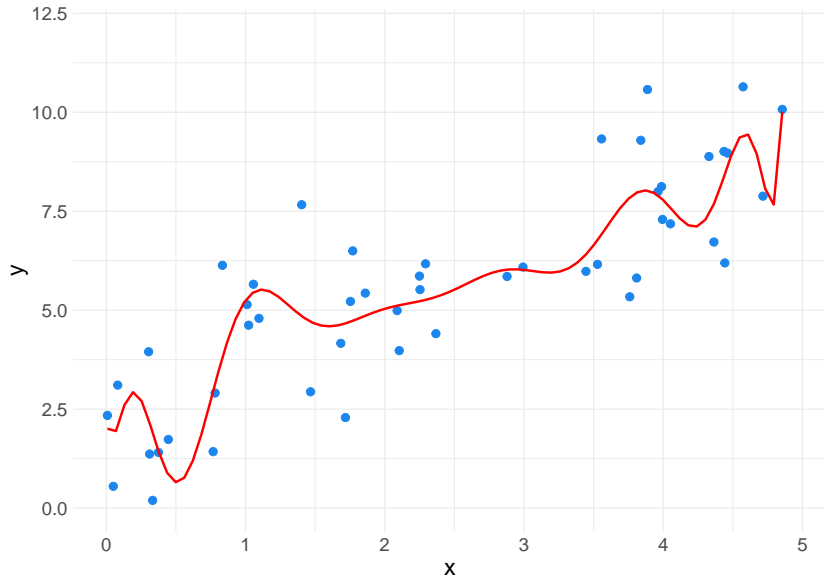


Out of sample performance of OLS

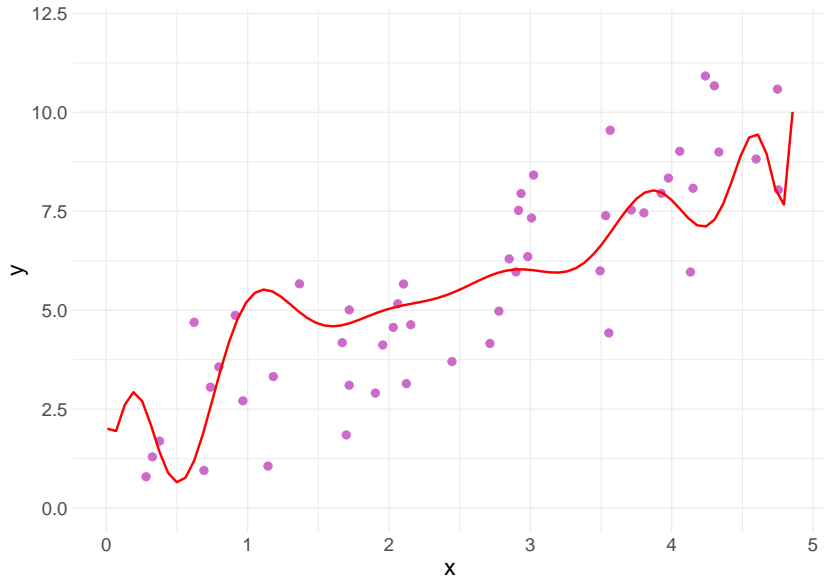
This trade-off explains why we might not want to use OLS for prediction tasks:

- ▶ By virtue of GM assumptions and B(L)UE, the MSE is explained entirely by the variance
 - ▶ Across models, the parameters will be centred on the true values (an unbiased estimator)
- ▶ We cannot tweak the model to get slightly better out-of-sample predictions at the expense of some added bias
 - ▶ In other words, we cannot leverage the bias-variance trade-off

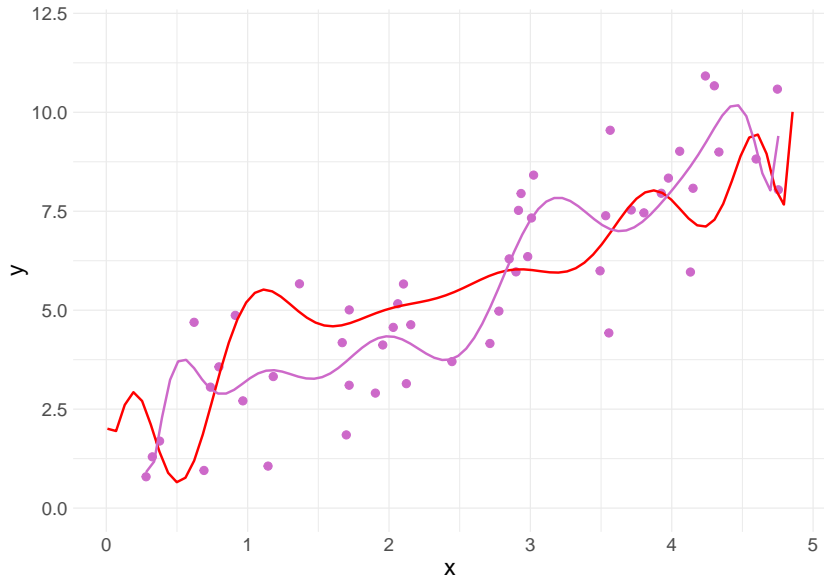
OLS: Complex OLS model trained on \mathbf{X}



OLS: Compare models' predictions to \mathbf{X}'



OLS: Variance in model predictions over X



The LASSO estimator

Regularisation and overfitting

In the previous examples, a complex model yields poor out-of-sample predictions.

To ensure our model does not have too much variance, we can *penalise* this complexity

- ▶ This will introduce bias into the model
- ▶ But, if done well, we can reduce the *total* MSE by offsetting overly-high variance
- ▶ And therefore yield better predictions on \mathbf{X}'

We call this process **regularisation**:

- ▶ Especially important where, by design, ML algorithms are highly flexible/complex
- ▶ Limits **overfitting**

Regularisation of OLS

If we want to regularise OLS we need to modify its loss function L :

- ▶ L_{OLS} is unbiased
- ▶ So, adding any non-zero term will *add* bias...
- ▶ ... and hopefully improve out-of-sample prediction

In other words:

- ▶ We sacrifice some variance in order to improve the predictive performance of the model on \mathbf{X}'

Generalising OLS with regularization

Following Kleinberg et al (2015) we can state this problem as the linear optimisation of:

$$\arg \min_f \underbrace{\sum_{i=1}^N (y_i - f(\mathbf{x}_i))^2}_{\text{Sum of squared error}} + \underbrace{\lambda R(f)}_{\text{Regularisation}}$$

For OLS:

- ▶ $f \in \mathcal{F}_{\text{linear}}$
- ▶ $\lambda = \frac{1}{\infty} = 0$

But what if $\lambda \neq 0$?

- ▶ Then we must decide what $R(\cdot)$ is
- ▶ And decide on a value of λ – a hyperparameter (more on selecting these tomorrow!)

$R(\cdot)$ as shrinkage

Consider an OLS model with k parameters:

- ▶ The model estimates coefficients for each parameter
- ▶ Regardless of how large or small that coefficient is
- ▶ In a sense, with non-zero estimates for each parameter these models can be considered “complex”

We can reduce the complexity of the regression model by setting some parameters to zero

- ▶ I.e. we **shrink** the coefficient estimates
- ▶ Aim to reduce the variance error by more than the increase in bias error

In the linear framework, we need some way to penalize non-zero coefficients

Least absolute shrinkage and selection operator (LASSO)

We can calculate the total magnitude (or **L1 Norm**) of the coefficients in a model as:

$$||\beta||_1 = \sum_j |\beta_j|$$

Next, we can think about restricting the size of this norm:

$$||\beta||_1 \leq t$$

Finally we can include this term in our loss function:

$$\arg \min_{\beta} \sum_{i=1}^N (y_i - \mathbf{x}_i \beta)^2 \text{ subject to } ||\beta||_1 \leq t,$$

which is equivalent to:

$$\arg \min_{\beta} \sum_{i=1}^N (y_i - \mathbf{x}_i \beta)^2 + \lambda ||\beta||_1.$$

LASSO

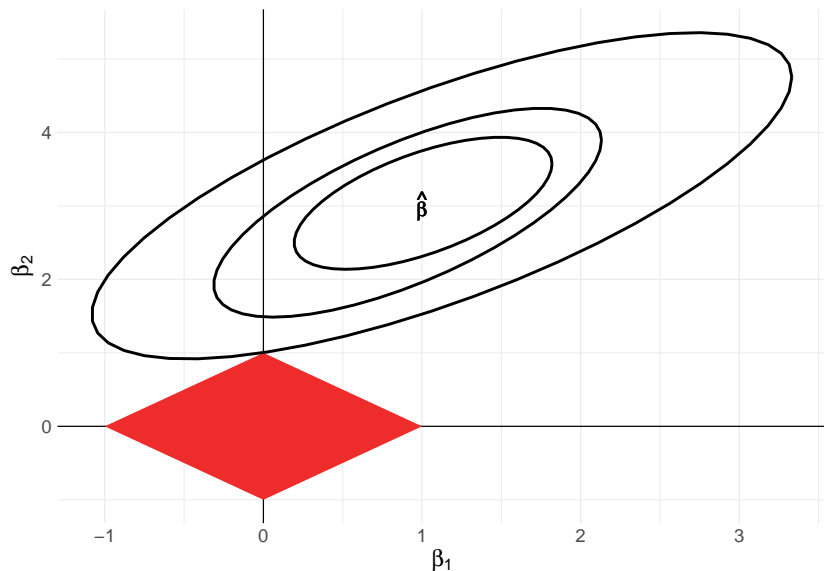
Hence, the LASSO estimator conforms to the generalised loss function introduced earlier

- ▶ $R(f) = ||\hat{\beta}||_1$

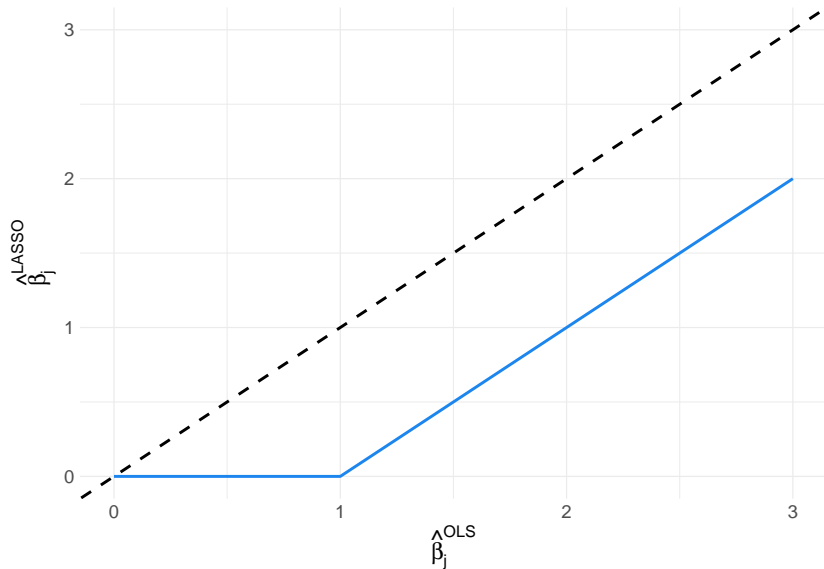
LASSO often yields coefficient estimates of exactly zero:

- ▶ Think about varying the true value of some coefficient β_j :
 - ▶ When β_j is large, we might shrink it (relative to OLS) but the importance of this predictor is sufficient to entail a non-zero coefficient
 - ▶ But for some small enough value b , the cost of including b in L1-norm is greater than the reduction in squared error
- ▶ In other words, the L1 norm constraint can lead to “corner solutions”

Example of LASSO corner solution ($||\beta||_1 \leq 1$)



Comparison of $\hat{\beta}_j^{\text{OLS}}$ to $\hat{\beta}_j^{\text{LASSO}}$



Two helpful properties of LASSO

1. Prediction accuracy

- ▶ We trade off an amount of bias for a (hopefully) greater reduction in variance, improving out-of-sample prediction
- ▶ Cf. a non-zero *true* coefficient estimated with a large confidence interval

2. Selection of relevant variables

- ▶ The possibility of corner-solutions acts as a useful variable selection mechanism
- ▶ LASSO essentially selects the most important variables for us

Application

Generalised applications

What sorts of general topics/data might we expect to see LASSO models?

Generalised applications

What sorts of general topics/data might we expect to see LASSO models?

- ▶ Mueller Hannes, Rauh Christopher. Reading Between the Lines: Prediction of Political Violence Using Newspaper Text. American Political Science Review. 2018;112(2):358-375.
 - ▶ Topic selection
- ▶ Kim In Song. Political Cleavages within Industry: Firm-level Lobbying for Trade Liberalization. American Political Science Review. 2017;111(1):1-20.
 - ▶ Word selection
- ▶ Gerring John, Jerzak Connort, Öncel Erzen. The Composition of Descriptive Representation. American Political Science Review. 2023:1-18.
 - ▶ Variable selection more generally

Blackwell and Olsen (2022)

Suppose we have an outcome y , a treatment d , covariates X , and an “effect moderator” v

- ▶ We want to estimate an *inference* model
- ▶ Understand how the treatment effect is moderated

Naive suggestion:

- ▶ Include an interaction term to model the differential effect of treatment
 - ▶ I.e. $y_i = \beta_0 + \beta_1 d_i + \beta_2 v_i + \beta_3 d_i v_i + \beta' X_i$

What could be wrong with this model?

Blackwell and Olsen (2022)

Suppose we have an outcome y , a treatment d , covariates X , and an “effect moderator” v

- ▶ We want to estimate an *inference* model
- ▶ Understand how the treatment effect is moderated

Naive suggestion:

- ▶ Include an interaction term to model the differential effect of treatment
 - ▶ I.e. $y_i = \beta_0 + \beta_1 d_i + \beta_2 v_i + \beta_3 d_i v_i + \beta' X_i$

What could be wrong with this model?

- ▶ We assume that the interactive effect β_3 is constant across covariates
- ▶ This introduces bias into the model if vX is related either to dv or y

Prediction and inference problem

Therefore the researcher faces a prediction problem *and* an inference problem:

- ▶ **Inference problem:** How do we control for potential bias introduced between X, v, d , and y ?
- ▶ **Prediction problem:** Which interactions within vX are most likely to confound the results?
 - ▶ Let us denote the true non-zero predictors \mathcal{P}
 - ▶ Inverting a \hat{y} problem – which variables are useful to predict new data?

From today's session we know that:

- ▶ Bias can be useful to offset variance when making out-of-sample predictions
- ▶ Bias inherently distorts our estimate of β

Combining LASSO and OLS

Blackwell and Olsen propose splitting the problem of interaction estimation into two stages:

1. **Variable selection**

- ▶ Use LASSO to estimate a series of variable selection models
- ▶ Attempt to find interaction terms that correlate with either outcome, treatment, or treatment-moderated interaction

2. **Inference**

- ▶ Use OLS to estimate an inference model
- ▶ Using only non-zero interaction terms in LASSO models

What makes this strategy so useful (and informative!) is that:

- ▶ We leverage bias to make better predictions in Stage 1
- ▶ We de-bias inference in Stage 2 using OLS + Stage 1 results

Post-double selection method

Stage 1

- ▶ Estimate LASSO models for:
 1. y on $\{v, X, vX\}$
 2. d on $\{v, X, vX\}$
 3. dv on $\{v, X, vX\}$
- ▶ Let Z^* index all variables with non-zero coefficients in any of models 1-3

Stage 2

- ▶ Regress y on d, dv and Z^*

Blackwell and Olson also suggest adding all “base-terms” (i.e X) regardless of LASSO coefficient

Coding workshop: Implementing post-double selection using LASSO

Extra Slides

Alternative $R(f)$ to the L1 norm

Following a similar logic to the shrinkage used by LASSO, we can define other measures of magnitude, like the L2 norm $||\beta||_2$:

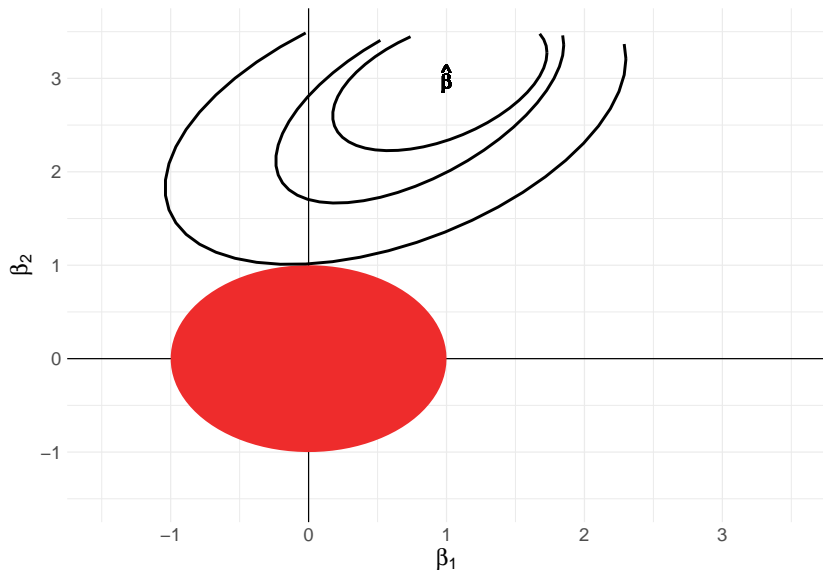
$$\sqrt{\sum_j |\beta_j|^2}$$

When we plug in the L2 norm into the loss function, we get the **ridge regression** estimator:

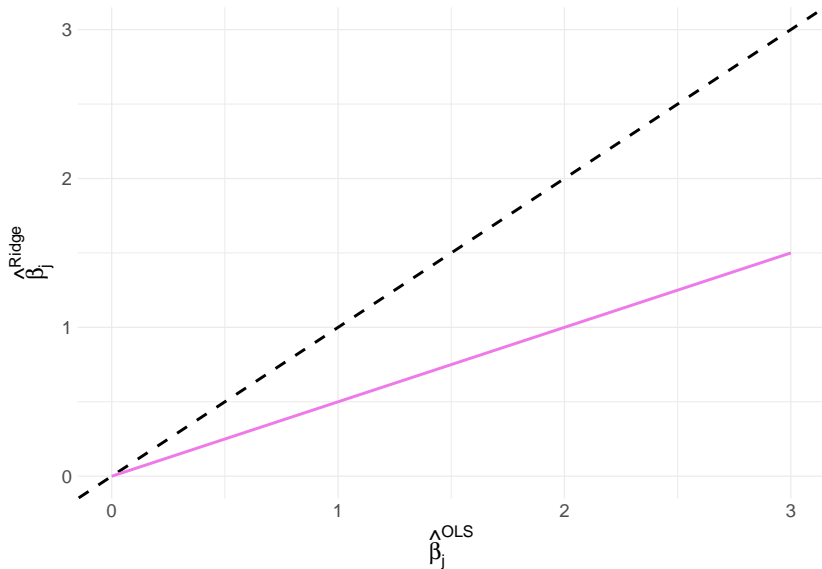
$$\arg \min_{\beta} \sum_{i=1}^N (y_i - \mathbf{x}_i \beta)^2 + \lambda ||\beta||_2$$

Unlike the LASSO estimator, ridge regression does not have a sharp cut-off, but rather scales the size of all coefficients in the model

Ridge regression – no corner solutions



Ridge regression – constant scaling of coefficients



Gauss Markov Assumptions

Five assumptions need to hold:

1. \mathbf{y} is a linear function of β
2. $\mathbb{E}[\epsilon_i] = 0$
3. $\mathbb{V}[\epsilon_i] = \sigma_i^2, \forall i$
4. $Cov(\epsilon_i, \epsilon_j) = 0, \forall i \neq j$
5. $Cov(\mathbf{x}_i, \epsilon_i) = 0$