Penalized Regression and Cross-Validation

Gov 51: Data Analysis and Politics

Scott Cunningham

Harvard University

Week 9 Thursday
March 27, 2026

When Experts Disagree: Data Science Meets the Courtroom

Andrew Baker is rewriting how courts use statistical evidence

- Assistant Professor, UC Berkeley School of Law
- PhD/JD from Stanford's joint business–law program
- Part of a new generation: attorneys with deep data science training

- Studies corporate finance, securities litigation, scientific integrity
- Fluent in machine learning, causal inference, and modern statistics
- If you're considering law school: this skill set barely existed 10 years ago

Here at Harvard Law, Crystal Yang does similar work — law + rigorous empirics

Courts rely on expert witnesses because judges aren't statisticians

- **The setup:** one side claims damages (securities fraud, antitrust, discrimination)
- **The judge's problem:** "How much did the defendant's actions cost the plaintiff?"
- **The solution:** hire an expert witness — a PhD social scientist or statistician who builds a model and testifies about the answer

Expert witnesses are hired guns with PhDs.
Each side picks one. They almost never agree.

Every modeling choice is a subjective decision

The expert must decide:

- Which variables go in the model?
- Which comparison group?
- Which time window?
- Which functional form?

- Each choice is defensible
- But different choices → different answers
- Plaintiff's expert picks choices that maximize damages
- Defendant's expert picks choices that minimize damages

The *researcher's* discretion — not the data — is driving the result

Baker's proposal: let the data pick the model

Baker (2025), *Berkeley Business Law Journal*

- Instead of letting experts choose variables subjectively ...
- Use penalized regression (LASSO, Ridge, Elastic Net)
- The **penalty** decides which variables stay and which go
- **Cross-validation** picks the penalty strength
- No expert discretion over model specification

The methods we learn today are Baker's tools.
First, let's understand *why* they work

Start with the Question: The Population Estimand

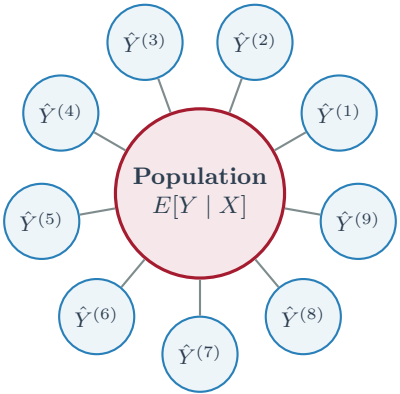
Every study starts with a question about the population — not the sample

- Did the fraud lower Halliburton's stock price *in the population of trading days*?
- Are Black defendants rearrested at higher rates *in the population of defendants*?
- Does education increase wages *for the population of workers*?

The **estimand** is the population quantity we want to know.
It has no randomness. It just *is*

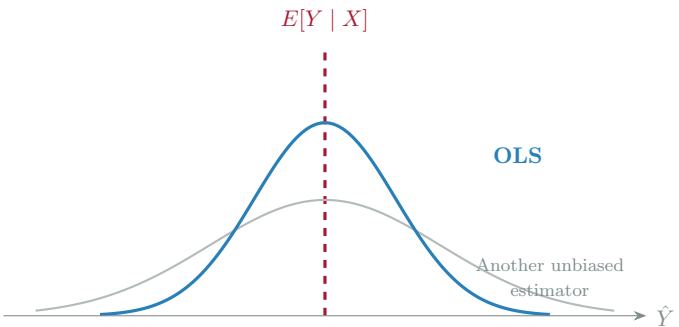
Our sample gives us an *estimate* of it — and that estimate varies across samples

$\hat{Y}$ is our sample's best guess at the population's $E[Y \mid X]$



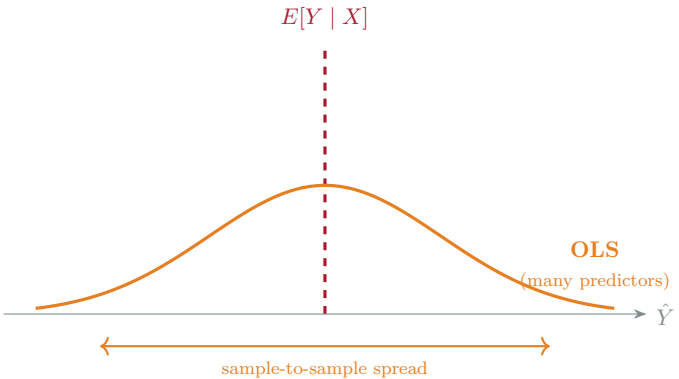
Each sample gives a different $\hat{Y}$. The hub — $E[Y \mid X]$ — never moves

OLS is centered on the truth — and Gauss-Markov says it's the tightest unbiased option



Both centered on the truth (unbiased).
OLS is the **narrowest** — Gauss-Markov guarantees this

But “narrowest among unbiased” can still be wide



- With many predictors, OLS fits each sample’s noise
- Centered on truth ✓ Tight? **Not at all**
- The MSE formula shows us why — and what to do about it

Review: The Bias-Variance Tradeoff

OLS minimizes training error — but MSE is about new data

OLS solves:

$$\min_{\beta} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \quad \text{— the } \textbf{training} \text{ data}$$

MSE asks:

$$E\left[(\hat{Y}_{new} - Y_{new})^2\right] \quad \text{— } \textbf{new} \text{ data you haven't seen}$$

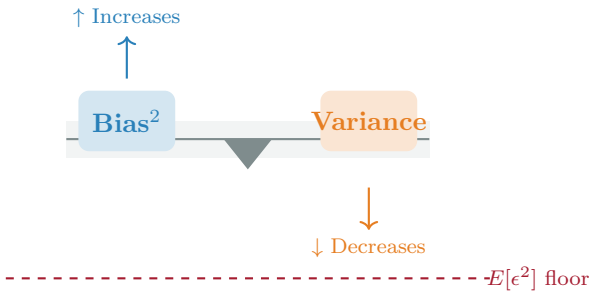
- Same objective function, **different data**
- OLS drives training residuals down by fitting noise
- That noise doesn't repeat in new data → overfitting

Every prediction error comes from bias, variance, or noise

$$E\left[(\hat{Y} - Y)^2\right] = \underbrace{\text{Bias}^2}_{\text{systematic error}} + \underbrace{\text{Variance}}_{\text{instability}} + \underbrace{E[\epsilon^2]}_{\text{irreducible noise}}$$

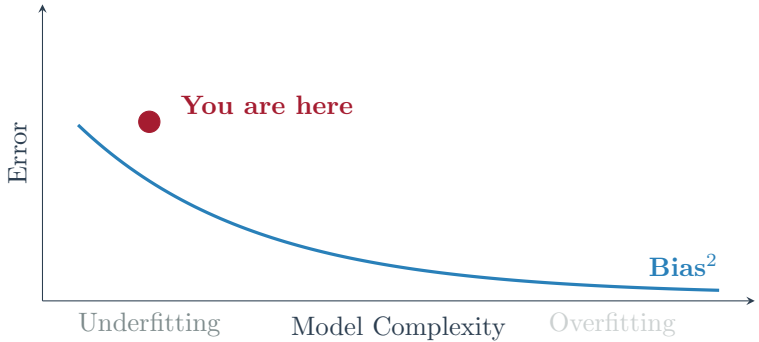
- We control bias and variance
- We cannot control $E[\epsilon^2]$ — that's the world's randomness
- The goal: minimize the **sum**, not either piece alone

You can trade bias for variance — but you can't escape the sum



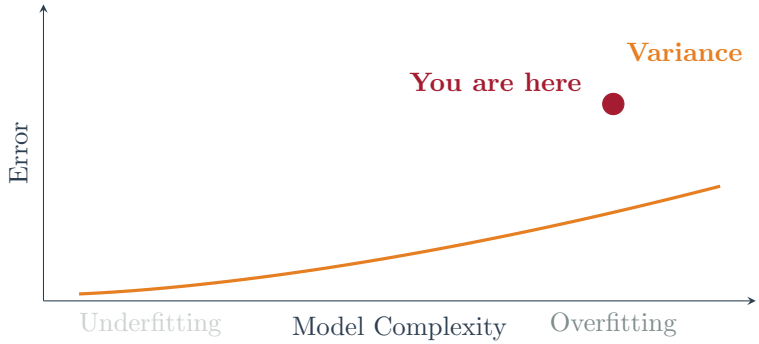
Noise is a floor. You can't go below it. Bias and variance trade off above it

Underfitting: high bias dominates



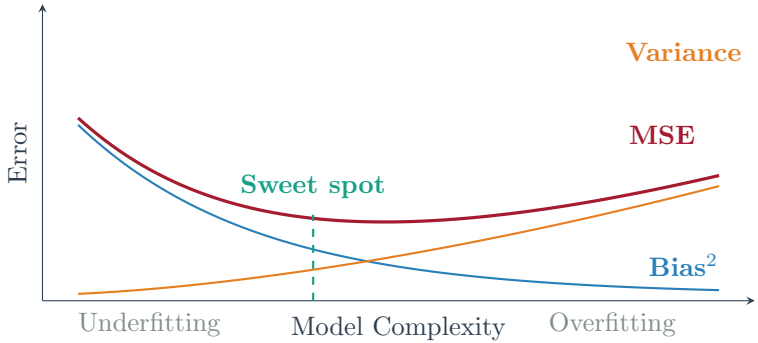
Simple model. Stable across samples. But systematically wrong.

Overfitting: high variance dominates



Complex model. Right on average, but wildly different each sample.

The sweet spot minimizes total error



We need a method that finds this sweet spot automatically

- Manually picking variables doesn't scale: $\binom{35}{20} = 3.2$ billion subsets
- Different researchers pick different variables → different answers
- No principled way to choose — until now

What if we kept all the variables but forced the model to be disciplined about how much weight each one gets?

The Big Idea: Penalize Complexity

What if we forced our regression coefficients to be smaller?

OLS has no constraints on coefficient size

Recall: OLS chooses $\hat{\beta}$ to minimize

$$\sum_{i=1}^{n} \left(Y_i - \hat{\alpha} - \sum_{j=1}^{p} \hat{\beta}_j X_{ij} \right)^2$$

- No limit on how large $\hat{\beta}_j$ can be
- With many predictors, OLS chases noise

Penalized regression is the mechanism to trade bias for variance

$$\min_{\beta} \underbrace{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}_{\text{fit the data}} + \underbrace{\lambda \cdot \text{Penalty}(\beta)}_{\text{keep coefficients small}}$$

- The penalty **prevents coefficients from reaching their OLS values**
- That introduces bias — but stabilizes estimates across samples
- $\lambda = 0$: no penalty → OLS (unbiased, high variance)
- $\lambda \to \infty$: all coefficients shrink toward zero (high bias, low variance)
- **Cross-validation** — testing each λ on held-out data — picks the right λ

Three penalized regression methods dominate applied work

- **Ridge** (Hoerl & Kennard, 1970) — shrinks all coefficients
- **LASSO** (Tibshirani, 1996) — shrinks and eliminates coefficients
- **Elastic Net** (Zou & Hastie, 2005) — blends both

Regularization goes beyond these three. Methods like random forests, XGBoost, and neural networks also discipline model complexity — just through different mechanisms. If you master the regression penalization methods, they will help you later when you encounter the others. We won't cover those in this class.

Ridge Regression (1970)

DuPont needed stable predictions to run a chemical plant

Arthur Hoerl and Robert Kennard were statisticians at DuPont's engineering lab in Wilmington, Delaware

- Their job: predict chemical properties from correlated measurements (temperature, pressure, concentration)
- **The problem:** correlated predictors made OLS wildly unstable — tiny changes in the data produced huge swings in coefficients
- Monday's model says "add 50 units of reagent X." Tuesday's says "subtract 30." You can't run a factory like that
- They needed predictions stable enough to bet real money on

Their fix: add a small penalty that prevents coefficients from exploding

OLS:	**Ridge:**
Coefficients go wherever they want	Coefficients pay a cost for being large
Correlated predictors → estimates swing wildly	Correlated predictors → estimates stay stable

Why "ridge"? Under the hood, OLS solves a system of equations involving the data. When predictors are correlated, that system is wobbly. Adding λ stabilizes it — like adding a spine to a flat surface. Hoerl borrowed the term from his earlier work on chemical response surfaces.

The profession objected: Ridge is biased

- Biased — but *stable*
- **Objection:** Gauss-Markov proves OLS is best!
- **Response:** BLUE = best *unbiased*
- Allow a little bias → beat OLS on total error

This is the bias-variance tradeoff in action

Ridge shrinks all coefficients but never eliminates any

$$\min_{\beta} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

- Penalty on the **sum of squared coefficients**
- Every variable stays in the model, just with smaller weight
- Never sets any coefficient *exactly* to zero

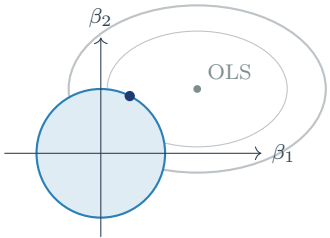
Why "L2"? The penalty measures coefficient size with squares

The L2 penalty:

$\lambda \sum_{j=1}^{p} \beta_j^2$

- Square each coefficient, add them up
- "L2" = the squared distance from zero
- Shrinks large coefficients aggressively
- Never reaches exactly zero

Geometrically:



Circle: no corners

LASSO (1996)

Ridge solved stability — but scientists wanted to know *which variables matter*

- Ridge keeps every variable, just with smaller coefficients
- 100 predictors in $\rightarrow$ 100 small coefficients out
- A doctor wants to know: which 5 genes actually predict this disease? Ridge can't tell you — it never drops anything
- **Robert Tibshirani** (Toronto, later Stanford) asked: what if the penalty could set coefficients to *exactly zero*?
- His insight: swap β_j^2 for $|\beta_j|$ — a tiny change in the math, a huge change in the result

Tibshirani invented LASSO at Toronto in 1996

Least **A**bsolute **S**hrinkage and **S**election **O**perator

- **Author:** Robert Tibshirani, Toronto $\rightarrow$ Stanford
- **Key move:** swap β_j^2 penalty for $|\beta_j|$ penalty
- **Result:** some coefficients shrink to **exactly zero**
- **LASSO selects**, not just shrinks

LASSO shrinks coefficients and sets some exactly to zero

$$\min_{\beta} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- Penalty on the **sum of absolute values**
- As λ increases, coefficients shrink
- Some hit exactly zero and drop out
- You get a sparse model: few variables, each one meaningful

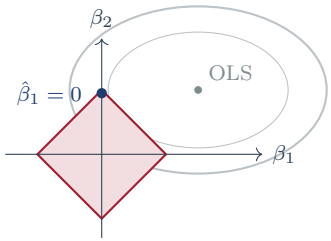
Why "L1"? The penalty measures coefficient size with absolute values

The L1 penalty:

$\lambda \sum_{j=1}^{p} |\beta_j|$

- Take the absolute value of each coefficient, add them up
- "L1" = the absolute distance from zero
- Shrinks *and* can hit exactly zero

Geometrically:



Diamond: corners on axes

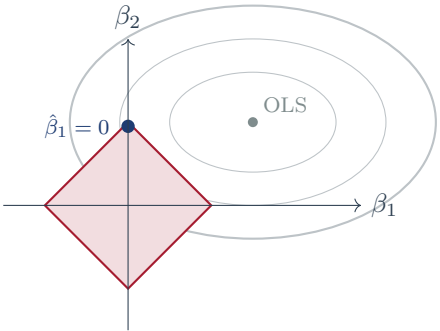
Corners → solutions land on axes → coefficients hit zero

LASSO tells you which variables the data actually needs

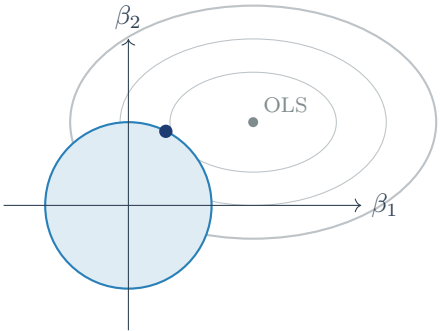
- **Prediction:** fewer noisy variables → less overfitting
- **Interpretation:** a short list of what matters
- **Honesty:** the data picks the model, not the analyst

The penalty selects variables → the analyst doesn't have to

The geometry explains why LASSO zeros out coefficients



LASSO (L1)

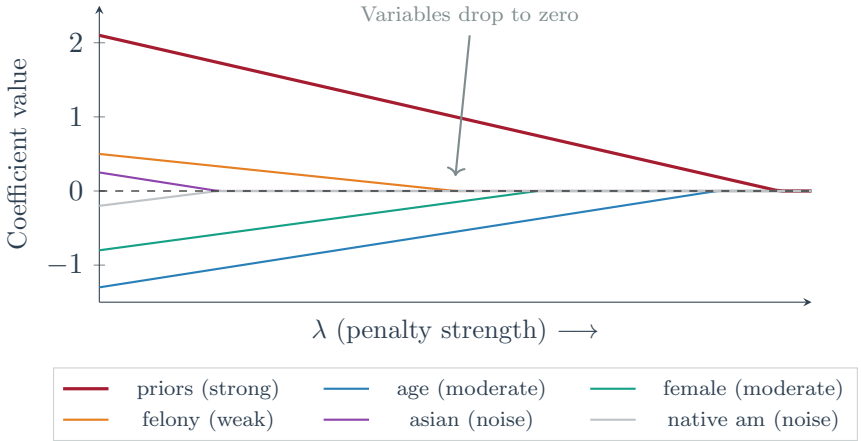


Ridge (L2)

Diamond corners → solutions land on axes → coefficients hit zero.

Circle has no corners → solutions never land exactly on an axis.

Prior record survives longest; noise variables drop out first



Elastic Net and the Full Picture

LASSO has a weakness: it's erratic when predictors are correlated

- If two predictors measure the same thing (e.g., height in inches and height in centimeters), LASSO arbitrarily keeps one and drops the other
- Run it again on a slightly different sample — it might keep the other one
- Ridge handles correlated predictors gracefully (shrinks both equally)
- **Hui Zou** (PhD student) and **Trevor Hastie** (advisor) at Stanford asked: can we get LASSO's variable selection *and* Ridge's stability?
- Their answer: blend both penalties — the "elastic net" stretches between the LASSO diamond and the Ridge circle

Elastic Net blends the LASSO and Ridge penalties

Zou & Hastie (2005)

$$\min_{\beta} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 + \lambda \Big[\alpha \sum_{j=1}^{p} |\beta_j| + (1 - \alpha) \sum_{j=1}^{p} \beta_j^2 \Big]$$

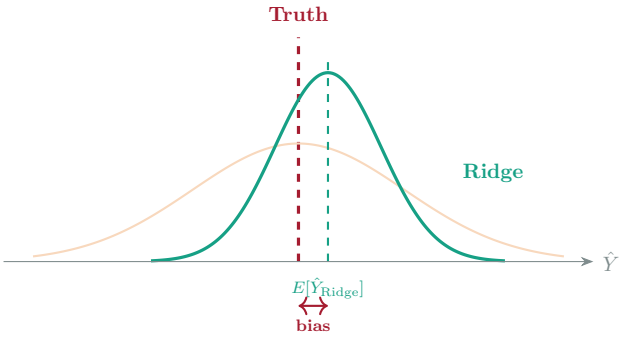
- $\alpha = 1$: pure LASSO
- $\alpha = 0$: pure Ridge
- $0 < \alpha < 1$: blend of both

Ridge, LASSO, and Elastic Net differ in how they shrink

	Ridge	LASSO	Elastic Net
Penalty	$\sum \beta_j^2$	$\sum \lvert\beta_j\rvert$	Both
Shrinks coefficients?	Yes	Yes	Yes
Sets coefficients to zero?	No	Yes	Yes
Variable selection?	No	Yes	Yes
Handles correlation?	Well	Poorly	Well

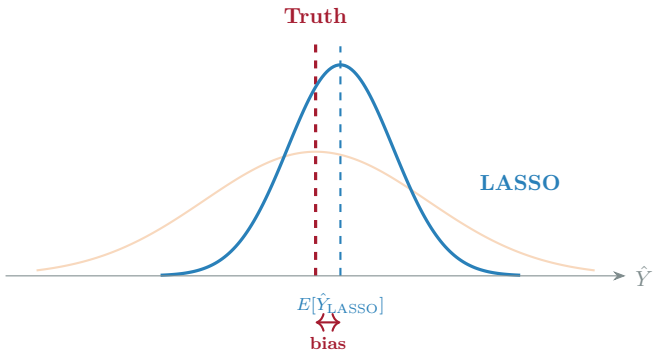
What Do These Methods Do to the Sampling Distribution?

Remember the wide OLS distribution? Here's what Ridge does to it



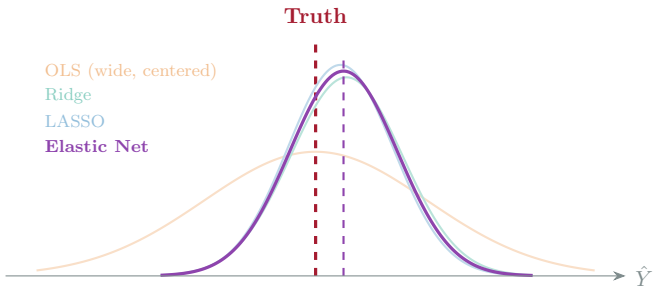
Narrower = lower variance. Small shift = small bias. Total MSE is *lower* than OLS

LASSO does the same thing — and kills noise variables



Same bias-variance tradeoff as Ridge — but LASSO also **selects variables**, setting noise predictors to exactly zero

All three penalized methods: tighter, slightly off-center



Every penalized method: *tighter and slightly off-center.*
The small bias is the price. The low variance is the payoff

How Do We Choose λ?

Back to crime prediction: which penalty strength works best?

LASSO for rearrest prediction: 30 candidate variables from admin data (age, prior arrests, charge type, time served, . . .)

- $\lambda = 0$: keep all 30 variables $\rightarrow$ OLS, overfits
- λ small: keep 22 variables, drop 8 weak ones
- λ medium: keep 9 variables, drop 21
- λ large: keep 1 variable (prior record), drop 29 $\rightarrow$ underfits

Which λ gives the most honest prediction of who gets rearrested?

The λ dial controls how much the model trusts the data

- λ **too small** — model trusts the data too much

  It memorizes quirks of the training sample. On new defendants, predictions are noisy.

- λ **too large** — model doesn't trust the data enough

  It ignores real patterns. Predictions are stable but systematically off.

- λ **just right** — the sweet spot

We need a method to **test** each candidate λ on data the model hasn't seen

How would you test λ by hand?

Imagine you have 500 defendants and want to predict rearrest:

1. Set aside the first 100 defendants — don't touch them yet
2. Fit your LASSO on the other 400, using some λ
3. Predict rearrest for the 100 you held out
4. Compute RMSE on those 100 predictions
5. Now put the first 100 back and hold out the next 100
6. Repeat: fit on 400, predict on 100, compute RMSE

After 5 rounds, every defendant has been in the "held-out" group exactly once

Each round's held-out group is called a **fold**. Five rounds = five folds = "5-fold cross-validation."

Try every λ, pick the one with the lowest average RMSE

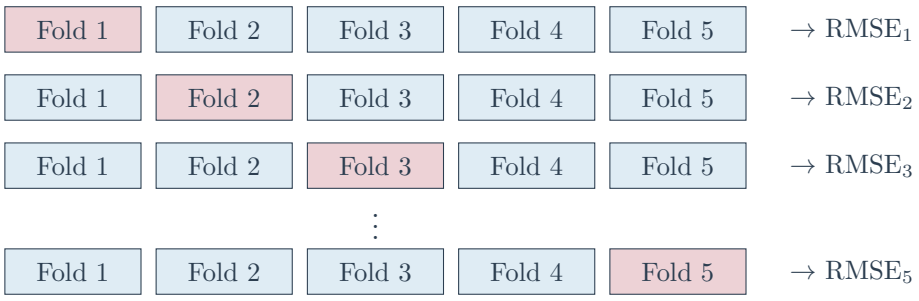
For each candidate λ:

1. Split data into 5 groups (folds)
2. Rotate through: each fold takes a turn as the test set
3. Average the 5 RMSE values $\rightarrow$ CV-RMSE for this λ

Then compare across λ values:

λ	Variables kept	CV-RMSE
0 (OLS)	30	0.48
0.01	22	0.44
0.05	**9**	**0.39 $\leftarrow$ best**
0.50	3	0.43
5.00	1	0.51

k-fold cross-validation: each observation gets a turn



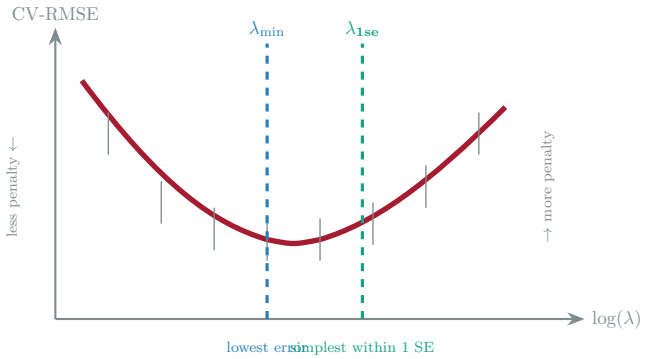
$$\text{CV-RMSE} = \frac{1}{5}(\text{RMSE}_1 + \cdots + \text{RMSE}_5)$$

Minimizing training error always picks $\lambda = 0$

- $\lambda = 0$ means no penalty $\rightarrow$ OLS
- OLS gives the best in-sample fit by definition
- But we already know OLS overfits!

Cross-validation estimates **out-of-sample** error $\Rightarrow$ optimal $\lambda > 0$

The CV error curve tells you which λ to use



- λ_{min}**:** lowest CV error — best raw prediction
- $\lambda_{\textbf{1se}}$**:** simplest model within 1 SE of minimum — fewer variables, nearly as good
- **Default:** use $\lambda_{1\text{se}}$ — simpler and almost as accurate

Two Cultures of Statistics

Breiman (2001): statistics has two cultures that barely talk

Data Modeling Culture	**Algorithmic Culture**
Specify a model *a priori*	Treat the process as unknown and complex
Estimate parameters (OLS)	Optimize prediction accuracy
Judge by R^2	Judge by out-of-sample error
Goal: understand the process	Goal: predict well
98% of statistics departments	Machine learning, industry

Before LASSO, the analyst picked the variables

- Researcher chooses variables based on theory and judgment
- Different researchers → different variable sets → different results
- Who's right? Whoever tells the most convincing story

After LASSO:

- Start with all candidate variables
- Let the penalty and cross-validation decide which ones stay

Back to Baker:
What Happened in Halliburton?

In Halliburton, different peer choices led to opposite conclusions

	Defense Expert	**Plaintiff Expert**
Peer selection	S&P Energy + custom index	Analyst-report peers
Excess return	-2.9%	-3.7%
p-value	0.20	0.02
Conclusion	Not significant	Significant

Stakes: class certification worth billions — hinged on peer selection

All three regularization methods converge on the same answer

- Baker applied LASSO, Ridge, and Elastic Net to 29 candidate peers
- LASSO kept 9, zeroed out 20

Method	Excess Return	p-value
LASSO	-3.2%	≈ 0.06
Ridge	-3.3%	≈ 0.06
Elastic Net	-3.2%	≈ 0.06

Result: data-driven answer falls between the two experts — all three methods agree

Penalized methods beat OLS when overfitting is the problem

	OLS	Ridge	LASSO	Elastic Net
Variables used	all 29	all 29 (shrunk)	9 of 29	12 of 29
Test RMSE	highest	lower	lowest	$\approx$ LASSO
Bias	zero	small	small	small
Variance	high	low	low	low

OLS has zero bias but high variance $\Rightarrow$ highest MSE
Penalized methods trade a little bias for a lot of stability

LASSO removes expert discretion: the data picks the variables

Baker's argument:

- Traditional expert testimony → subjective variable selection → dueling experts
- LASSO + cross-validation → data-driven selection → convergence

For us:

- Same penalized regression we just learned
- Applied to a $5+ billion securities case

A skeptic would say: biased estimators give the wrong answer

The objection:

- Gauss-Markov proves OLS is best
- Introducing bias means our coefficients are *systematically wrong*
- How can "wrong on purpose" be better?

The response:

- "Best" among *unbiased* estimators $\neq$ best overall
- MSE = Bias^2 + Variance
- A tiny bias that crushes variance *lowers* total error
- In the Halliburton case: biased methods *converged* while unbiased experts *disagreed*

How Uncertain Is Each Prediction?

The judge doesn't just want the best model — she wants to know how sure it is

Bail decision for defendant #4,872:

- **Point prediction:** 23% chance of rearrest
- **But how uncertain is "23%"?**
- Could easily be 10% or 40% — those imply different decisions

RMSE tells you how the model does *on average*
A prediction interval tells you how
uncertain *this specific prediction* is

Confidence intervals and prediction intervals answer different questions

Confidence interval

$\hat{Y} \pm t \times \text{SE}_{\text{mean}}$

- Where is the **average** Y for people like this?
- Shrinks with more data
- Estimation uncertainty only

Prediction interval

$\hat{Y} \pm t \times \text{SE}_{\text{pred}}$

- Where will **this person's** Y actually land?
- Cannot shrink below noise floor
- Estimation uncertainty + **noise**

$$\text{SE}^2_{\text{pred}} = \text{SE}^2_{\text{mean}} + \widehat{\sigma}^2$$ — the extra $\widehat{\sigma}^2$ is irreducible noise from the decomposition

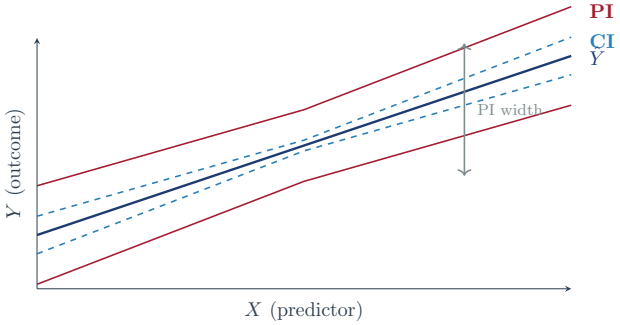
The PI formula has three terms — each connects to something you know

$$\hat{Y}_0 \;\pm\; t \times \sqrt{\widehat{\text{MSE}} \times \left(\underbrace{1}_{\text{noise}} + \underbrace{\frac{1}{n}}_{\text{intercept}} + \underbrace{\frac{(x_0 - \bar{x})^2}{\sum (x_i - \bar{x})^2}}_{\text{slope / leverage}} \right)}$$

Term	What it captures	Connection
1	New person's ε (iid draw)	Irreducible noise
$1/n$	Uncertainty about intercept	Shrinks with n — like $\text{SE}_{\bar{Y}}$
$(x_0 - \bar{x})^2 / \cdots$	Uncertainty about slope	Extrapolation penalty

Even with $n \to \infty$, the **1** never goes away — that's what "irreducible" means

The prediction interval is always wider than the confidence interval



- **CI** (dashed): narrows with more data — estimation uncertainty
- **PI** (solid): has a **floor** — irreducible noise never goes away

In R: one argument gives you prediction intervals

Confidence interval (for the mean):

```
predict(fit, newdata, interval = "confidence")
```

Prediction interval (for an individual):

```
predict(fit, newdata, interval = "prediction")
```

	fit	lwr	upr
CI for mean	\$1,117	\$1,095	\$1,139
PI for individual	\$1,117	\$542	\$1,692

Same point prediction. CI width: \$44. PI width: \$1,150.

For LASSO and Ridge, prediction intervals are an open problem

The problem:

- OLS has a clean PI formula
- LASSO/Ridge are biased by design
- Standard formula doesn't apply
- No `interval = "prediction"` in `glmnet`

What you can do:

- **RMSE:** average prediction error
- **Bootstrap:** resample, refit, predict → spread = uncertainty
- **Conformal prediction:** works for any model (Rambachan, MIT)

For your projects: OLS prediction intervals where possible, RMSE comparison across methods always

What's Next

Problem Set 3 is due Thursday, April 2

PS3: Prediction and Regularization

- COMPAS recidivism dataset (ProPublica)
- Fit OLS, then expand to ∼141 interaction terms
- Watch OLS overfit
- Fix it with Ridge, LASSO, and Elastic Net
- Cross-validation by hand, then with `cv.glmnet()`
- Discuss algorithmic fairness

Next week: same regression, different purpose

Week 10: From prediction to causal inference

- We've been asking: what will happen?
- Next: why did it happen?
- Covariates reduce MSE in prediction
- Covariates reduce *bias* in causal inference

The data should pick the model, not the analyst.

A little bias buys a lot of stability.