Machine Learning:
An Applied Econometric Approach

Jann Spiess

based on work with Sendhil Mullainathan

in collaboration with Susan Athey and Niall Keleher

2. The Secret Sauce of Machine Learning

Structure of first chapter of webinar

1. Introduction

Training data (y, x)	$\xrightarrow{\hat{f}}$	Application data $(\hat{y} = \hat{f}(x), x)$

2. The Secret Sauce of Machine Learning

3. Prediction vs Estimation

Prediction problem set-up

Given:

- Training data set $(y_1, x_1), \ldots, (y_n, x_n)$ (assume iid)
  - Usually called "regression" when y continuous, "classification" when y discrete
- Loss function $\ell(\hat{y}, y)$

	Econometrics	ML
y	Outcome variable	Label
x	Covariate	Feature

Goal:

- Prediction function $\hat{f}$ with low average loss ("risk")

$$\mathrm{L}(\hat{f}) = E_{(y,x)}\left[\ell\left(\hat{f}(x), y\right)\right]$$

where (y, x) distributed same as training

Squared-error loss for regression

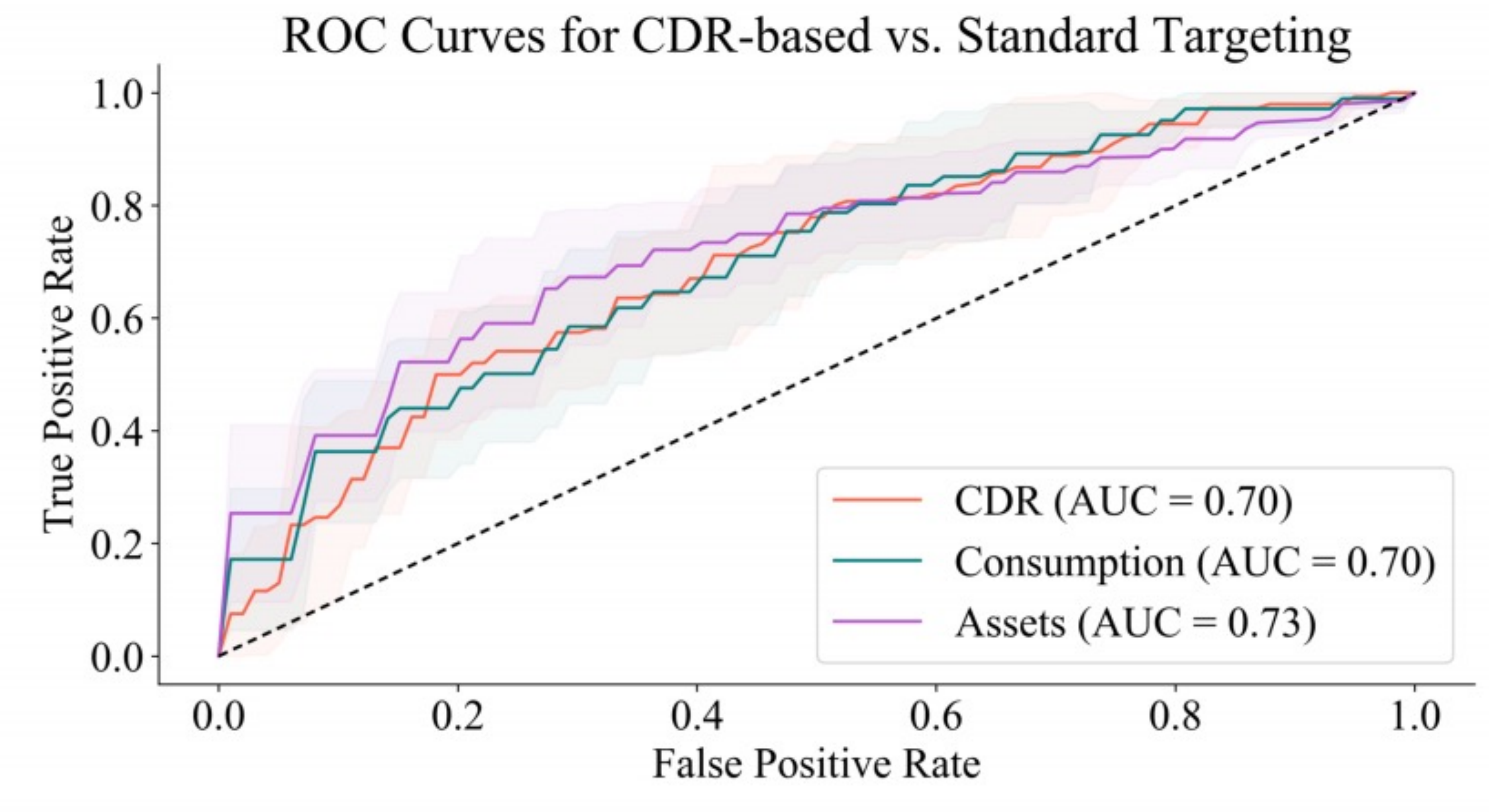
"Regression": Continuous outcome, $y \in \mathbb{R}$

$$\text{Squared-error loss: } L(\hat{f}) = E\left(\hat{f}(x) - y\right)^2$$
$$(\ell(\hat{y}, y) = (\hat{y} - y)^2)$$

- Predict log house price y of a new home from its characteristics x based on survey data from homes with same distribution (Mullainathan and Spiess, 2017)

- Predict log consumption y for a new household x based on data on similar households (Adelman et al.)

Loss measures for classification



ROC curve source: Aiken et al. (2020)

Standard regression solution

Goal: small $E\left(\hat{f}(x) - y\right)^2$

E.g. use linear functions $\hat{f}(x) = \hat{\beta}'x = \hat{\beta}_0 + \sum_{j=1}^{k} \hat{\beta}_j x_j$

- From training data, pick the β that provides best in-sample fit:

$$\min_{\hat{\beta}} E\left(y - \hat{\beta}'x\right)^2 \quad \rightarrow \quad \min_{\hat{\beta}} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\beta}'x_i\right)^2$$

- Which optimality properties does OLS have?
- Is this optimal for prediction?

Bias–variance decomposition

- Loss at new point $y = \beta' x + \epsilon$ ($E[\epsilon|x] = 0$):

$$(\hat{y} - y)^2 = \left(\hat{\beta}' x - \beta' x - \epsilon\right)^2$$

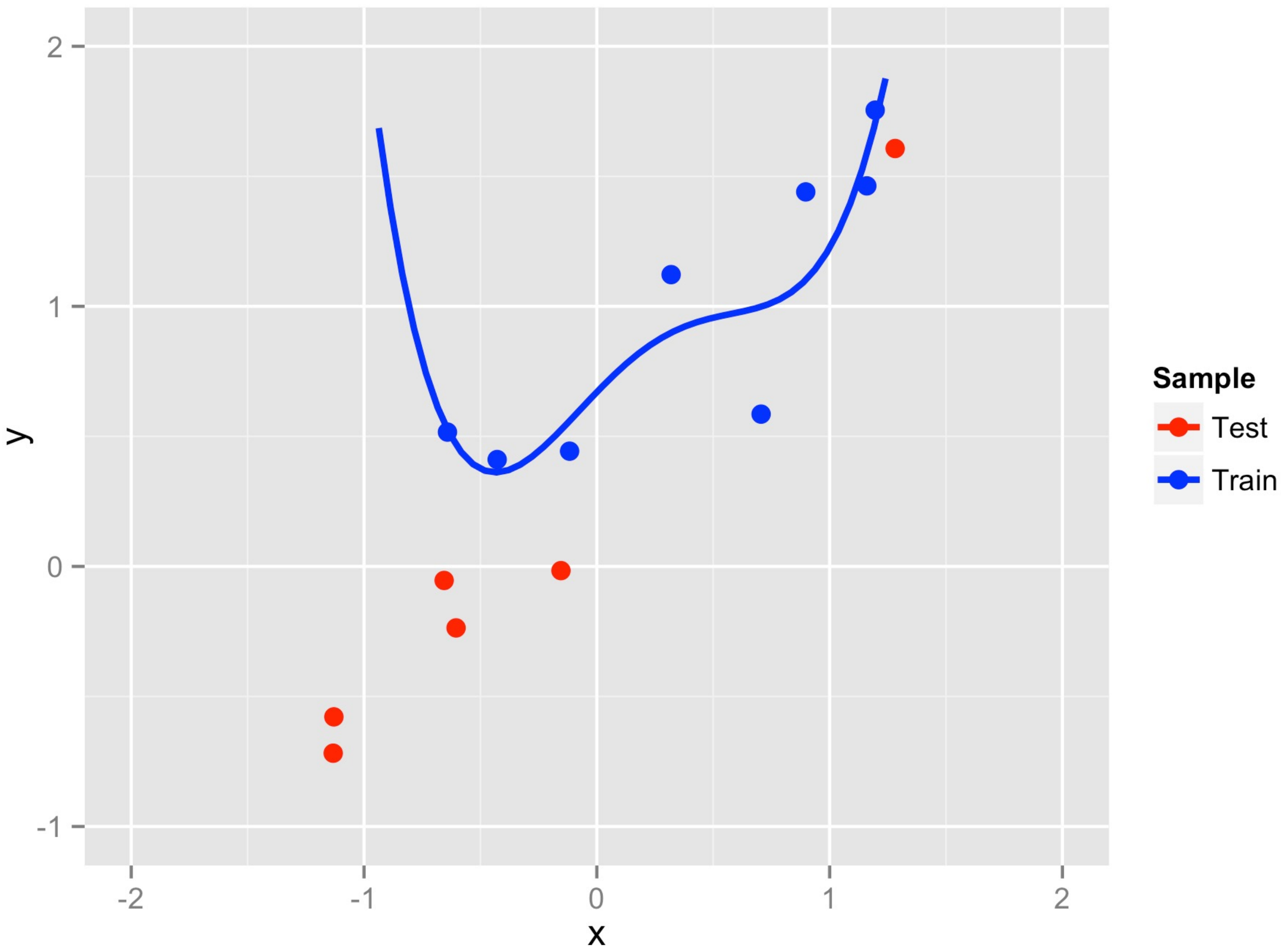
- Average over draws of training sample (and ϵ):

$$E_{T,\epsilon}[(\hat{y} - y)^2] = E_T\left[\left(\hat{\beta}' x - \beta' x\right)^2\right] + E_\epsilon[\epsilon^2]$$

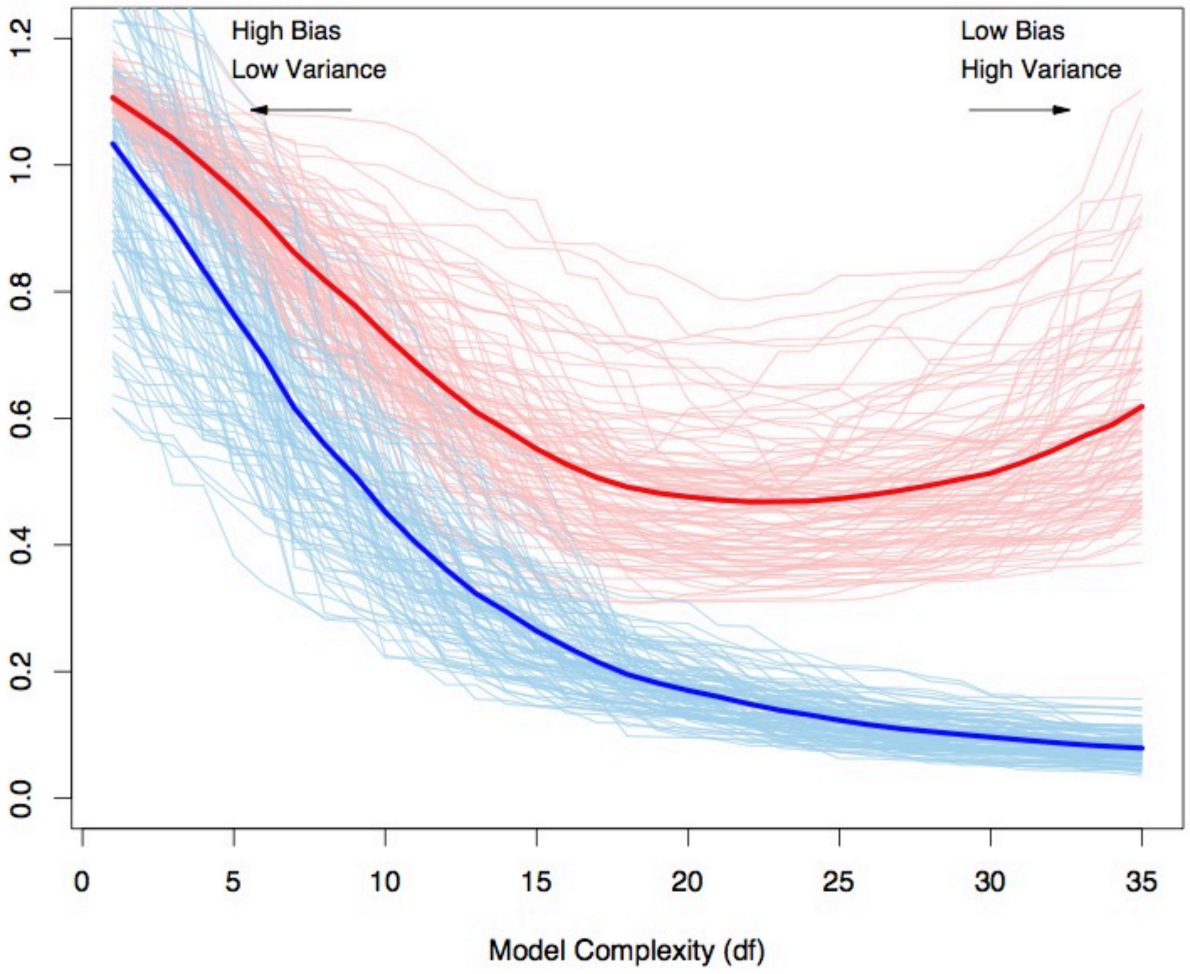
$$= \underbrace{\left(\left(E_T[\hat{\beta}] - \beta\right)' x\right)^2}_{\substack{\text{bias} \\ \text{approximation}}} + \underbrace{x' V_T(\hat{\beta}) x}_{\substack{\text{variance} \\ \text{overfit}}} + \underbrace{V_\epsilon(\epsilon|x)}_{\text{irreducible noise}}$$

- Important framing within econometrics, stats

Approximation–overfit trade-off



Approximation–overfit trade-off



Approximation–overfit trade-off

As model becomes more complex:

1. Fit true function better (approximation)
2. Fit noise better (overfit)

Hence:

1. Flexible functional forms
2. **Limit expressiveness (regularization)**

Regularization for linear regression

- Rather than OLS

$$\min_{\hat{\beta}} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\beta}' x_i\right)^2$$

- Fit constrained problem

$$\min_{\hat{\beta}} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\beta}' x_i\right)^2 \text{ s.t. } \left\|\hat{\beta}\right\| \leq c$$

$$\left\|\hat{\beta}\right\|_0 = \sum_{j=1}^{k} 1_{\hat{\beta}_j \neq 0} \qquad \left\|\hat{\beta}\right\|_1 = \sum_{j=1}^{k} |\hat{\beta}_j| \qquad \left\|\hat{\beta}\right\|_2^2 = \sum_{j=1}^{k} \hat{\beta}_j^2$$

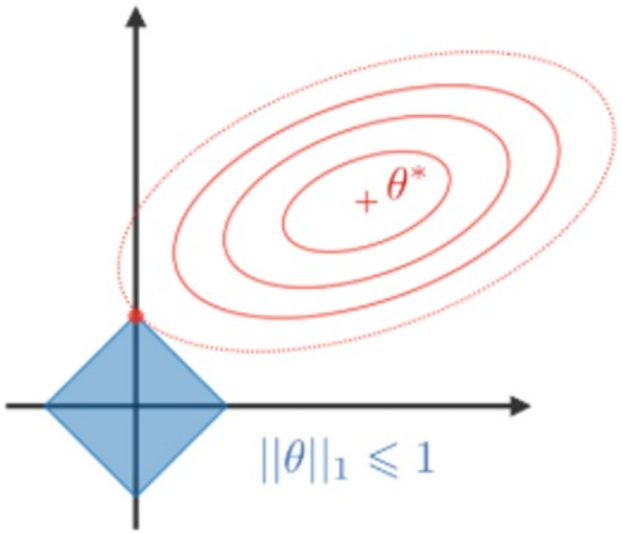
- Throughout, assume $\hat{\beta}' = (\boxed{\hat{\beta}_0}, \hat{\beta}_1, \dots, \hat{\beta}_k)$
- Normalize!

not penalized

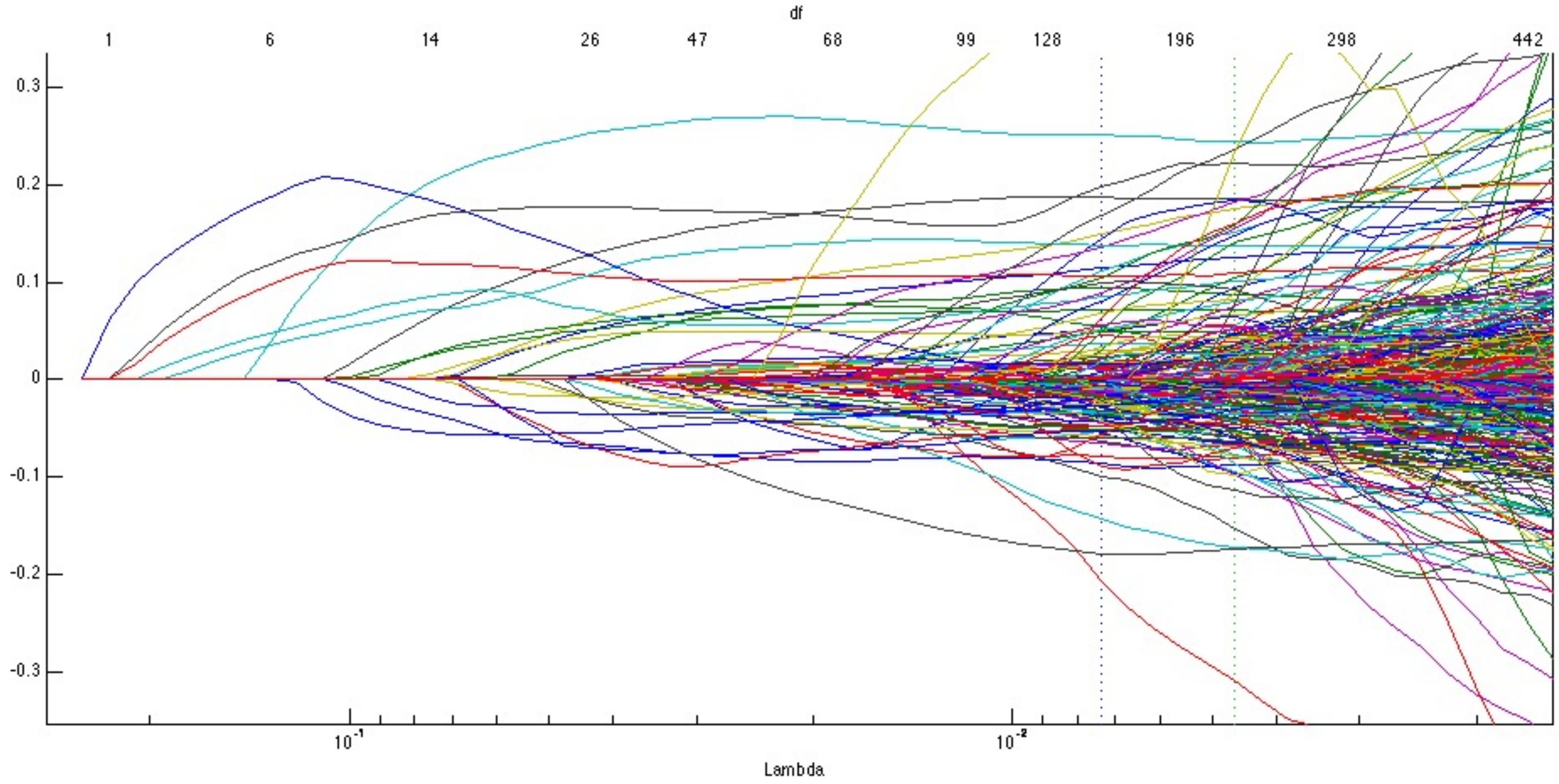
LASSO regression

$$\min_{\hat{\beta}} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\beta}' x_i\right)^2 + \lambda \sum_{j=1}^{k} |\hat{\beta}_j|$$

- Selects *and* shrinks
- "Capitalist" – in doubt give all to one
- Produces sparse solutions



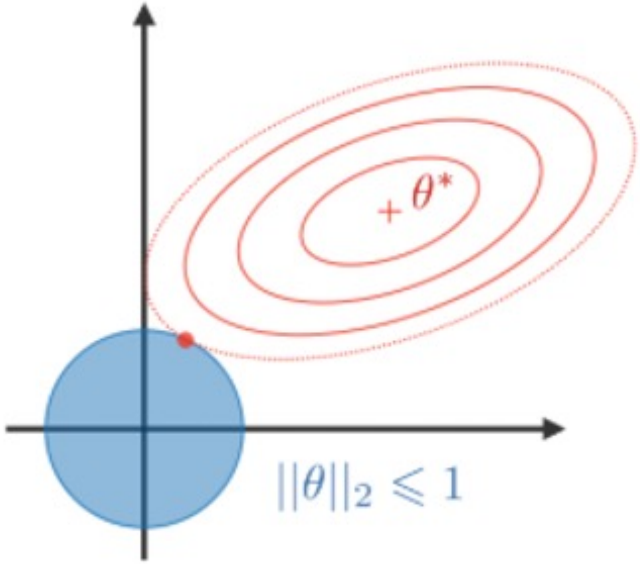
LASSO regression



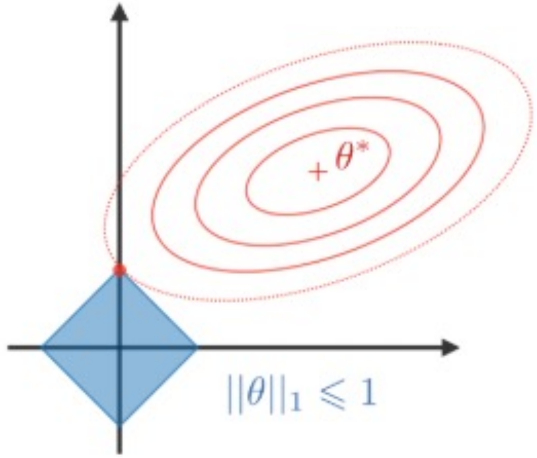
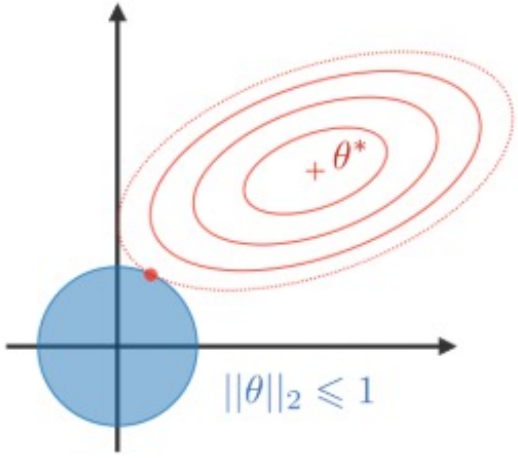
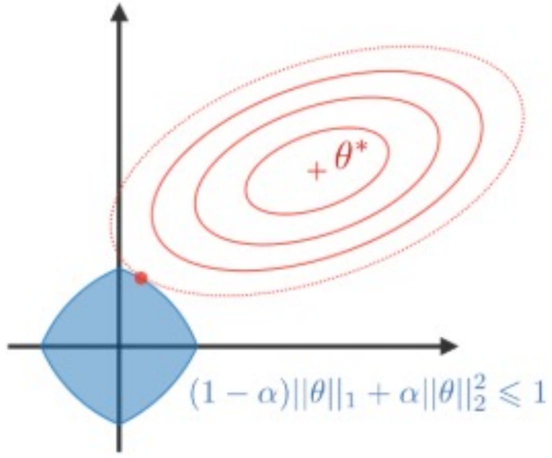
Ridge regression

$$\min_{\hat{\beta}} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\beta}' x_i\right)^2 + \lambda \sum_{j=1}^{k} \hat{\beta}_j^2$$

- Shrink towards zero, but never quite
- “Socialist” – in doubt distribute to multiple
- Can be interpreted as Bayesian posterior



Regularization for linear regression

LASSO	Ridge	Elastic Net
• Shrinks coefficients to 0 • Good for variable selection	Makes coefficients smaller	Tradeoff between variable selection and small coefficients
		
$... + \lambda\|\|\theta\|\|_1$ $\lambda \in \mathbb{R}$	$... + \lambda\|\|\theta\|\|_2^2$ $\lambda \in \mathbb{R}$	$... + \lambda\Big[(1-\alpha)\|\|\theta\|\|_1 + \alpha\|\|\theta\|\|_2^2\Big]$ $\lambda \in \mathbb{R}, \alpha \in [0,1]$

Source: Afshine Amidi and Shervine Amidi

Structure of supervised learners

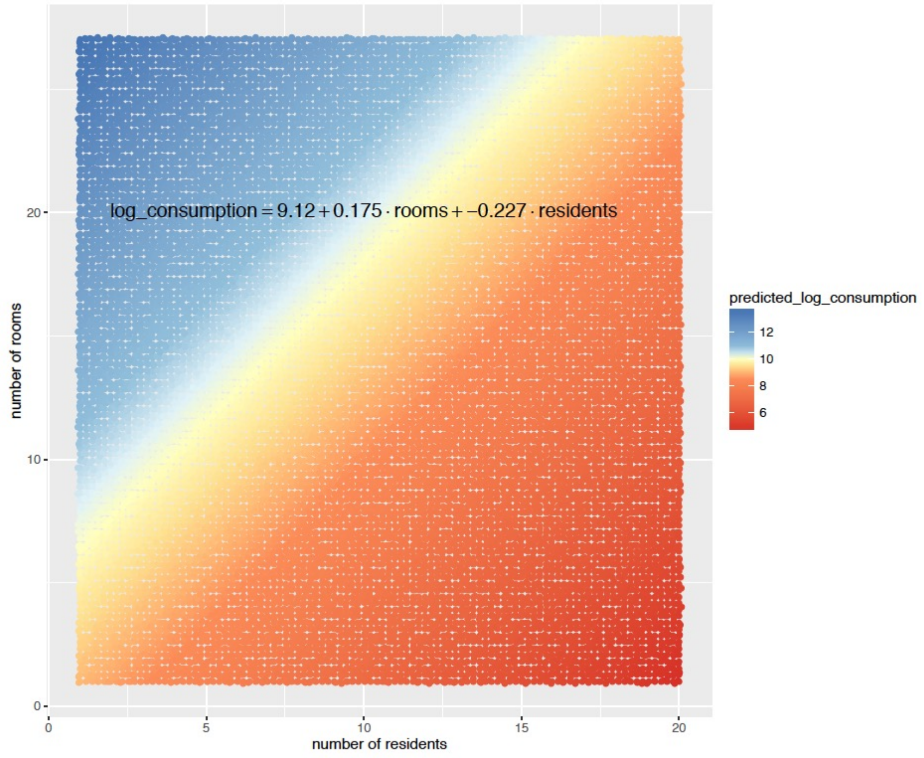
- A function class
- A regularizer
- An optimization algorithm that gets us there

Poverty targeting

Indicator	Value	Points	Scor
1. How many members does the household have?	A. Five or more	0	
	B. Four	6	
	C. Three	11	
	D. Two	17	
	E. One	20	
2. Do any household members ages 5 to 18 go to private school or private pre-school?	A. No	0	
	B. Yes	5	
	C. No members ages 5 to 18	7	
3. How many years of schooling has the female head/spouse completed?	A. Three or less	0	
	B. Four to eleven	2	
	C. Twelve or more	8	
	D. No female head/spouse	8	
4. How many household members work as employees with a written contract, as civil servants for the government, or in the military?	A. None	0	
	B. One	4	
	C. Two or more	13	
5. In their main occupation, how many household members are managers, administrators, professionals in the arts and sciences, mid-level technicians, or clerks?	A. None	0	
	B. One or more	8	
6. How many rooms does the residence have?	A. One to four	0	
	B. Five	2	
	C. Six	5	
	D. Seven	7	
	E. Eight or more	11	
7. How does the household dispose of sewage?	A. Ditch, other, or no bathroom	0	
	B. Simple hole, or directly into river, lake, or ocean	2	
	C. Septic tank not connected to public sewage/rainwater system	3	
	D. Septic tank connected to public sewage/rainwater system	4	
	E. Direct connection to public sewage/rainwater system	5	
8. Does the household have a refrigerator?	A. No	0	
	B. Yes, with one door	5	
	C. Yes, with two doors	10	
9. Does the household have a washing machine?	A. No	0	
	B. Yes	7	
10. Does the household have a cellular or land-line telephone?	A. None	0	
	B. Cellular but not land-line	5	
	C. Land-line but not cellular	6	
	D. Both	11	

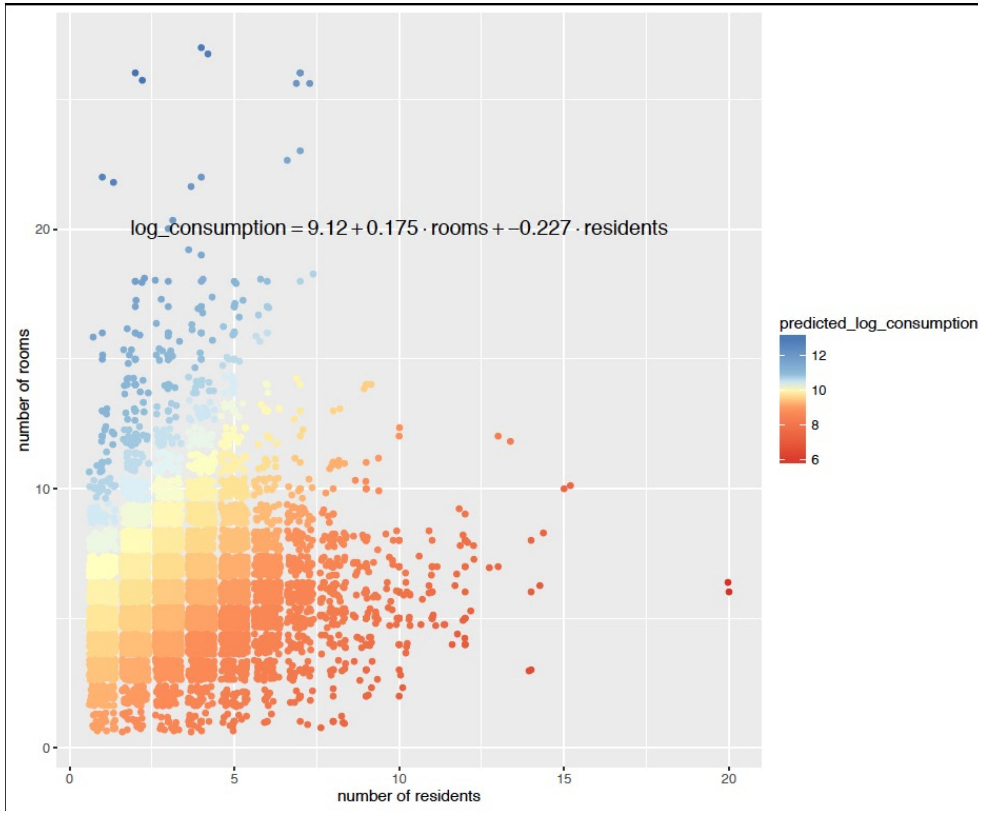
Example source: Adelman et al.

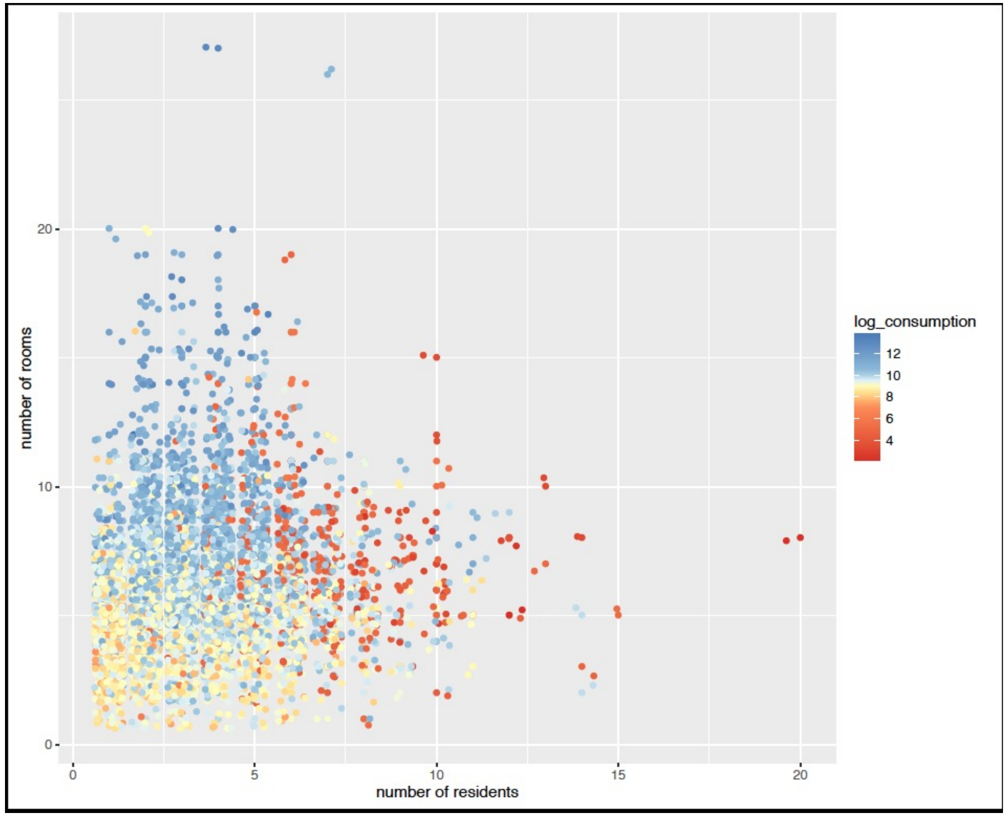
Reference point: OLS



Example source: Adelman et al.

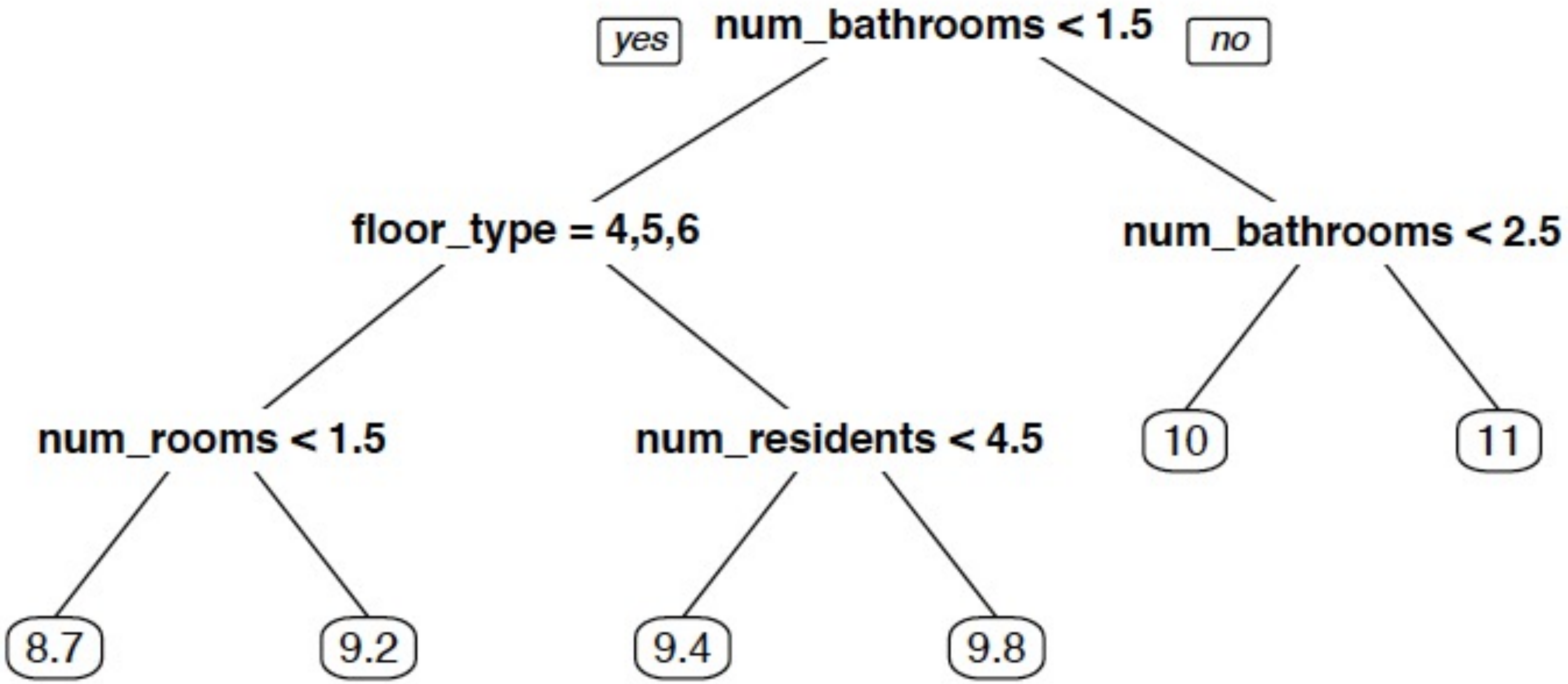
Fitted vs actual values in sample





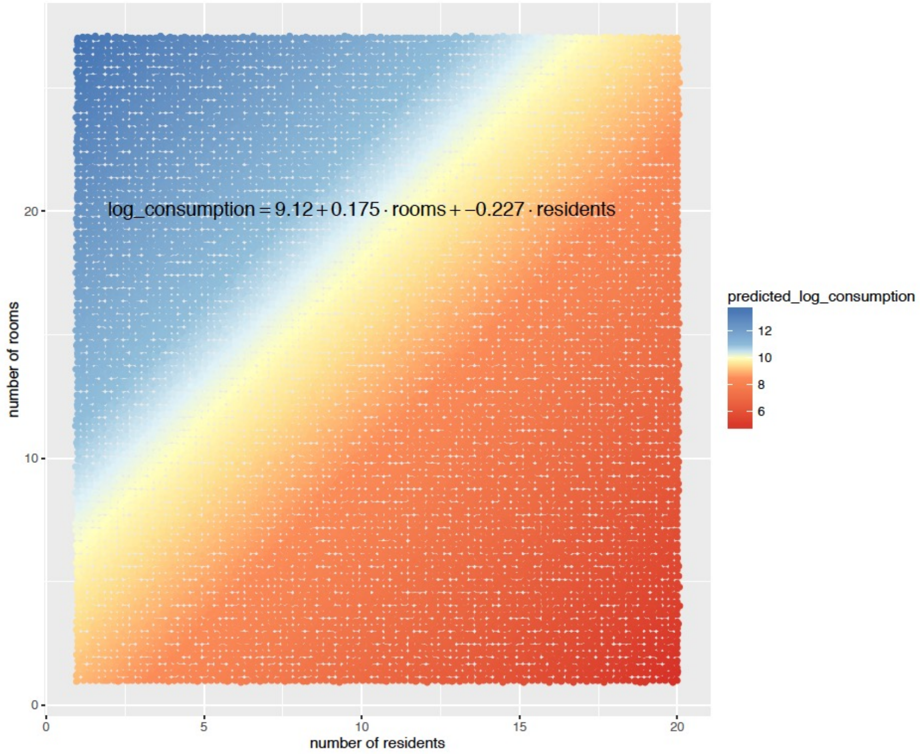
Example source: Adelman et al.

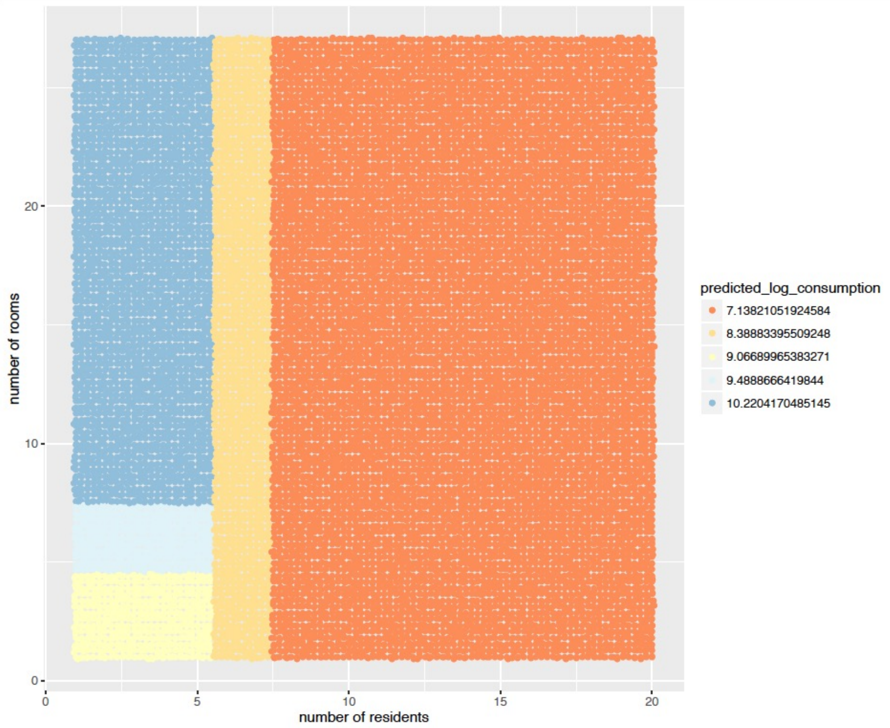
Regression trees



Example source: Adelman et al.

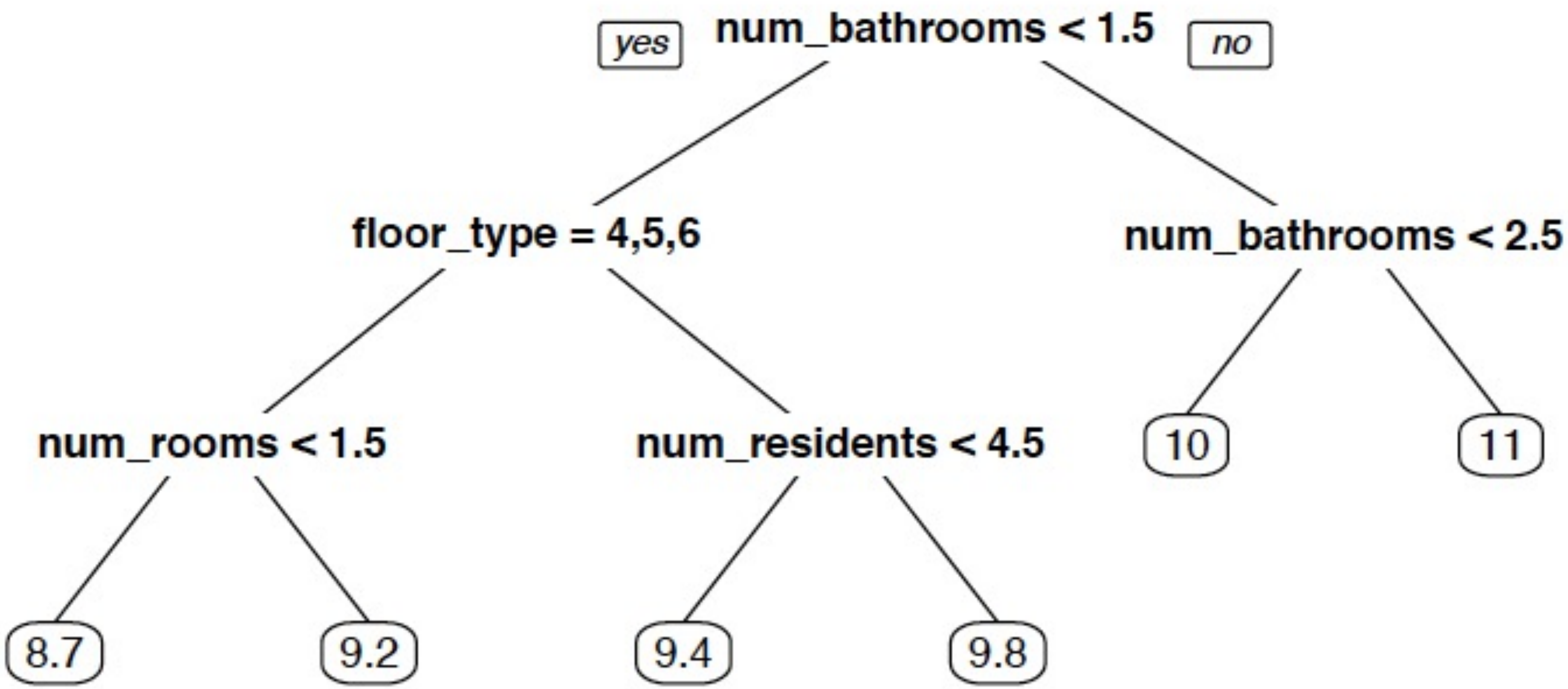
OLS vs tree





Example source: Adelman et al.

How to find optimal tree?

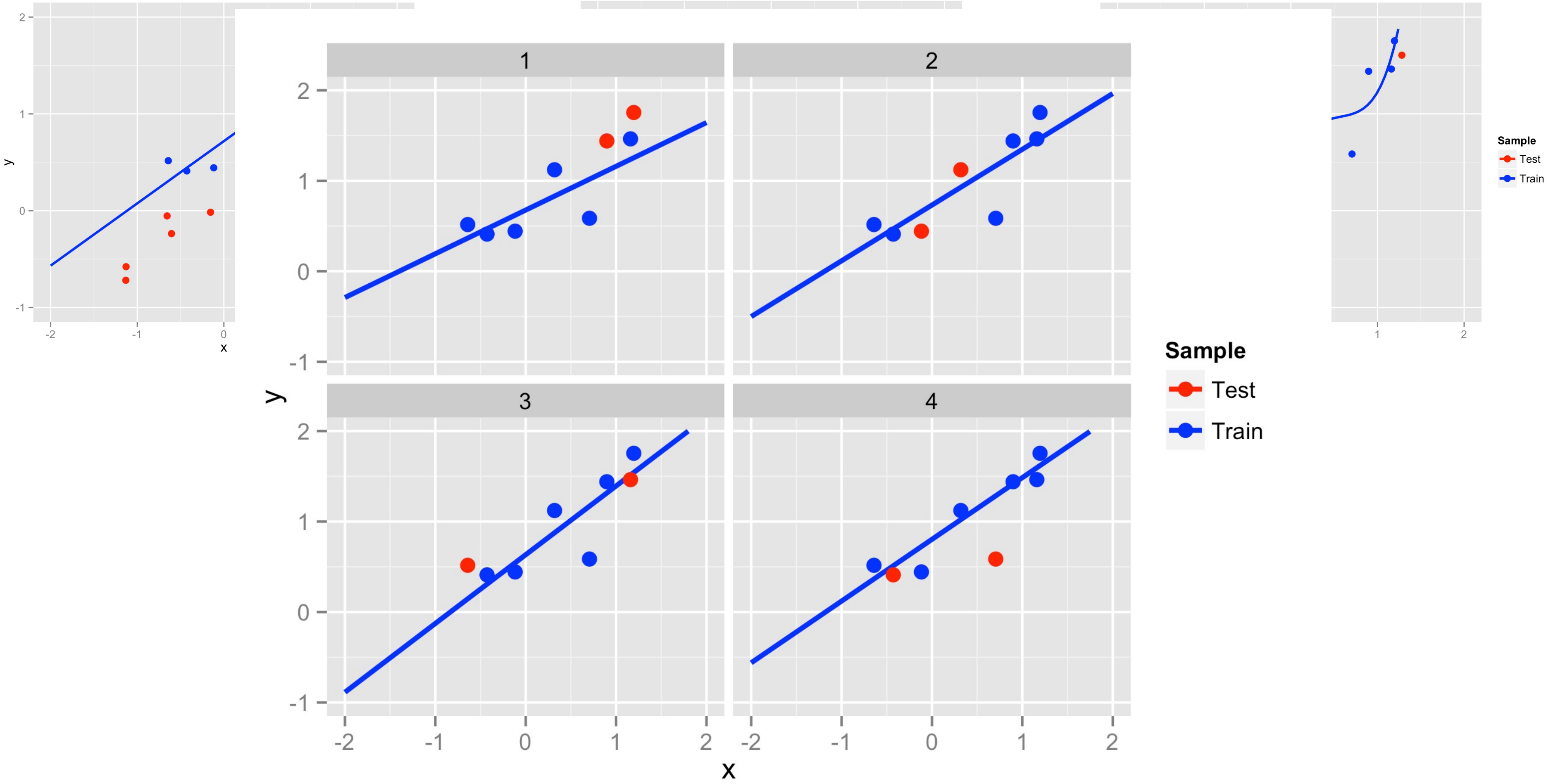


Example source: Adelman et al.

Structure of supervised learners

- A function class
- A regularizer
- An optimization algorithm that gets us there

Choosing regularization parameter



Choosing regularization parameter

- Hold-out: create out-of-sample in-sample
- Cross-validation: create repeated hold-outs

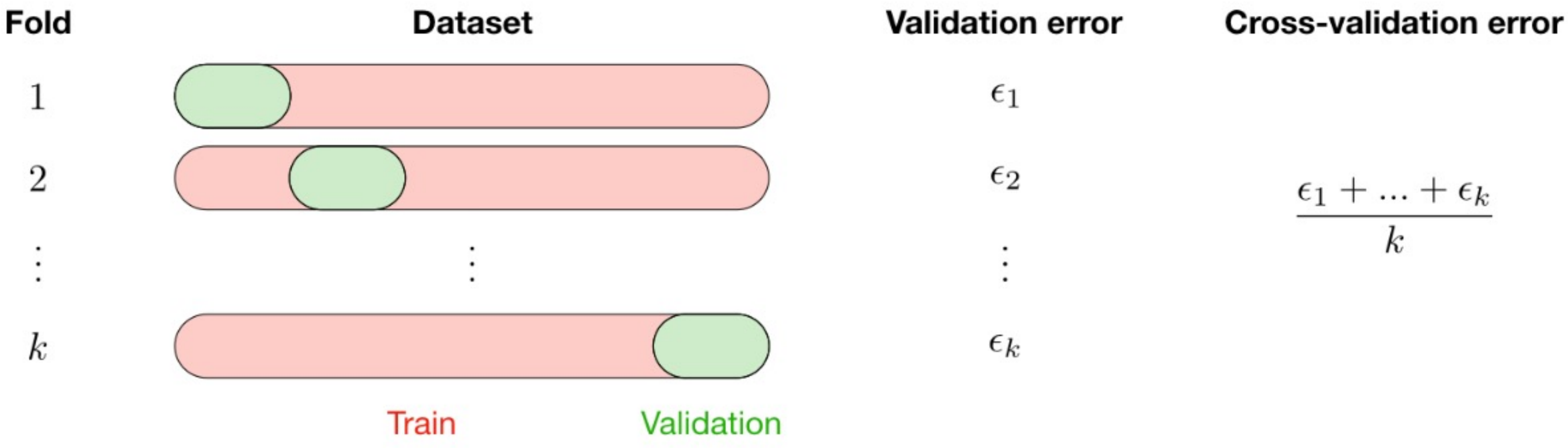


Illustration: Afshine Amidi and Shervine Amidi

Choosing regularization parameter

- Hold-out: create out-of-sample in-sample
- Cross-validation: create repeated hold-outs

Hence:

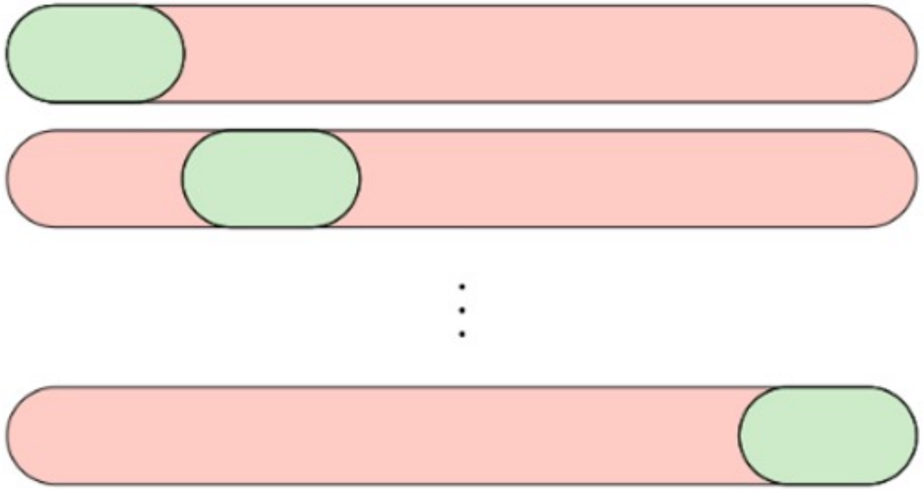
1. Flexible functional forms
2. Limit expressiveness (regularization)
3. **Learn how much to regularize (tuning)**

Structure of ML exercise

fitting sample | hold-out sample

engineering with econometric guidance

econometric guarantees



Obtain a function $\hat{f}$

Estimate $\mathbf{L}(\hat{f})$

firewall principle

ML basics recap

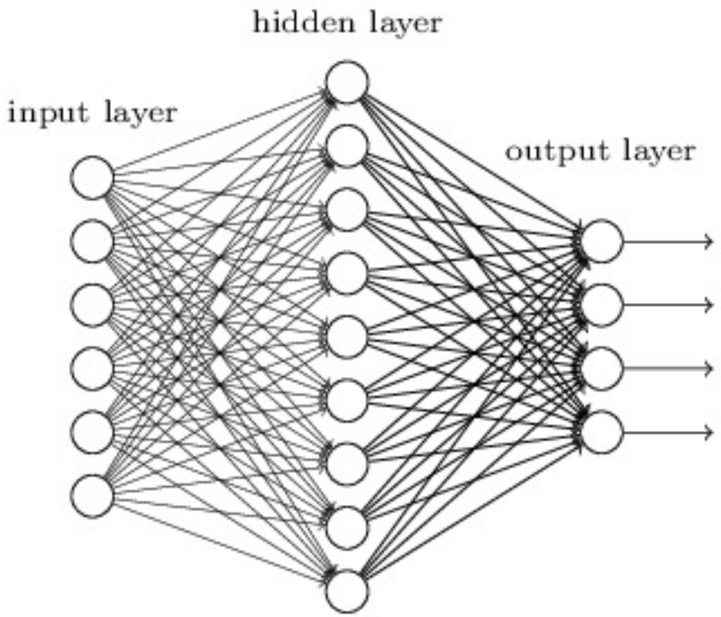
1. Flexible functional forms
2. Limit expressiveness (regularization)
3. Learn how much to regularize (tuning)

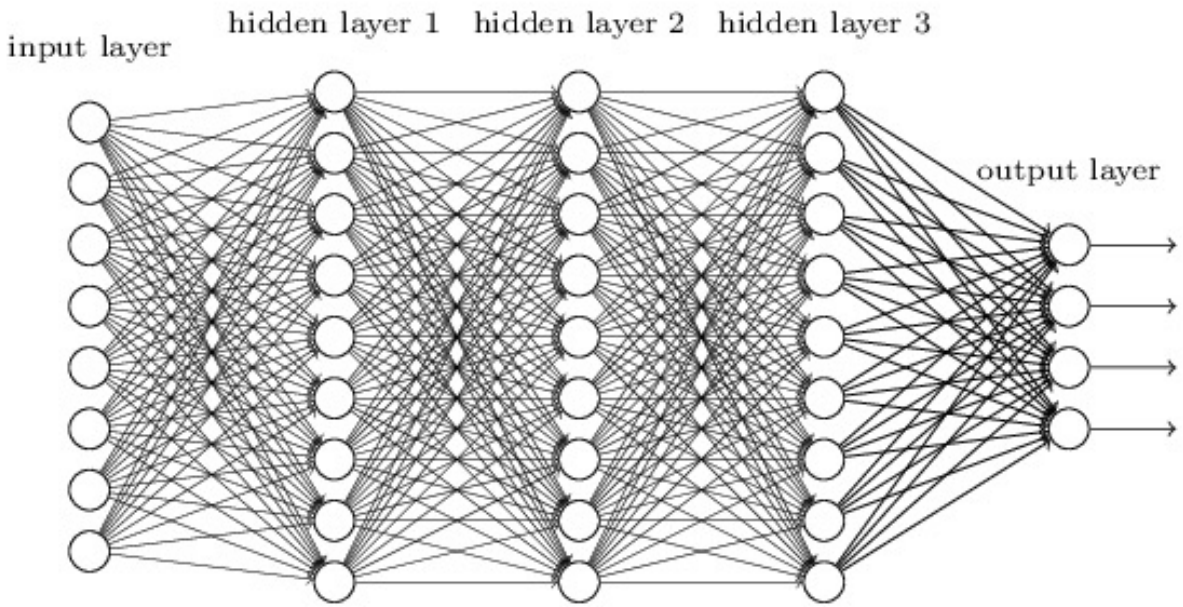
- Important researcher choices:
  - Loss function
  - Data management/splitting
  - Feature representation
  - Function class and regularizer

From LASSO to neural nets

Function class	Regularizer
Linear	LASSO, ridge, elastic net
Decision/regression trees	Depth, leaves, leaf size, info gain
Random forest	Trees, variables per tree, sample sizes, complexity
Nearest neighbors	Number of neighbors
Kernel regression	Bandwidth
Splines	Number of knots, order
Neural nets	Layers, sizes, connectivity, drop-out, early stopping

Regularizing neural nets



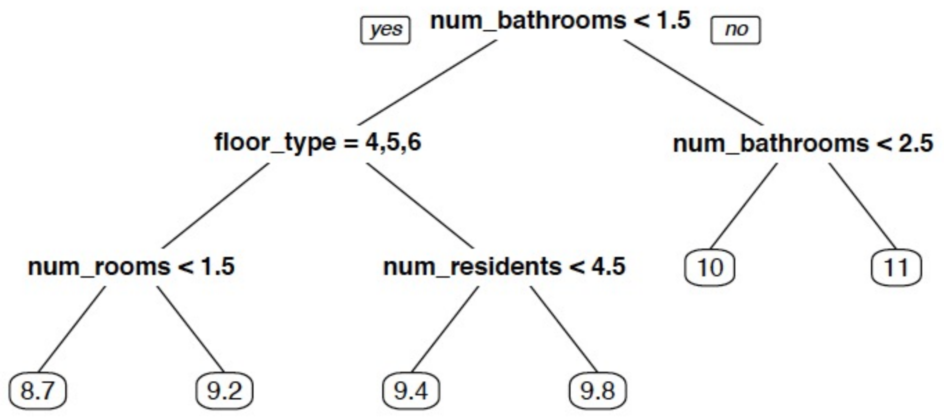


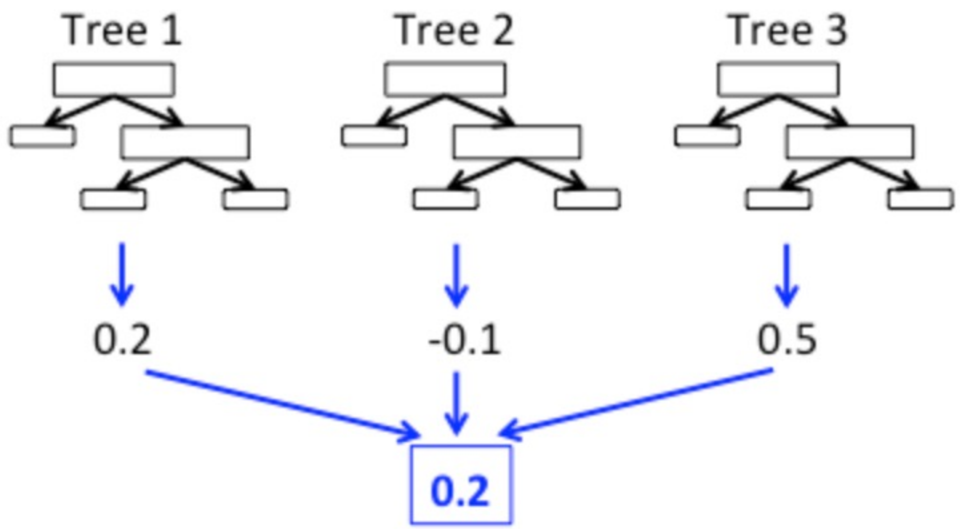
Model combination: ensembles

$$\hat{f}(x) = w_1\,\hat{f}_1(x) + \cdots + w_K\,\hat{f}_K(x)$$

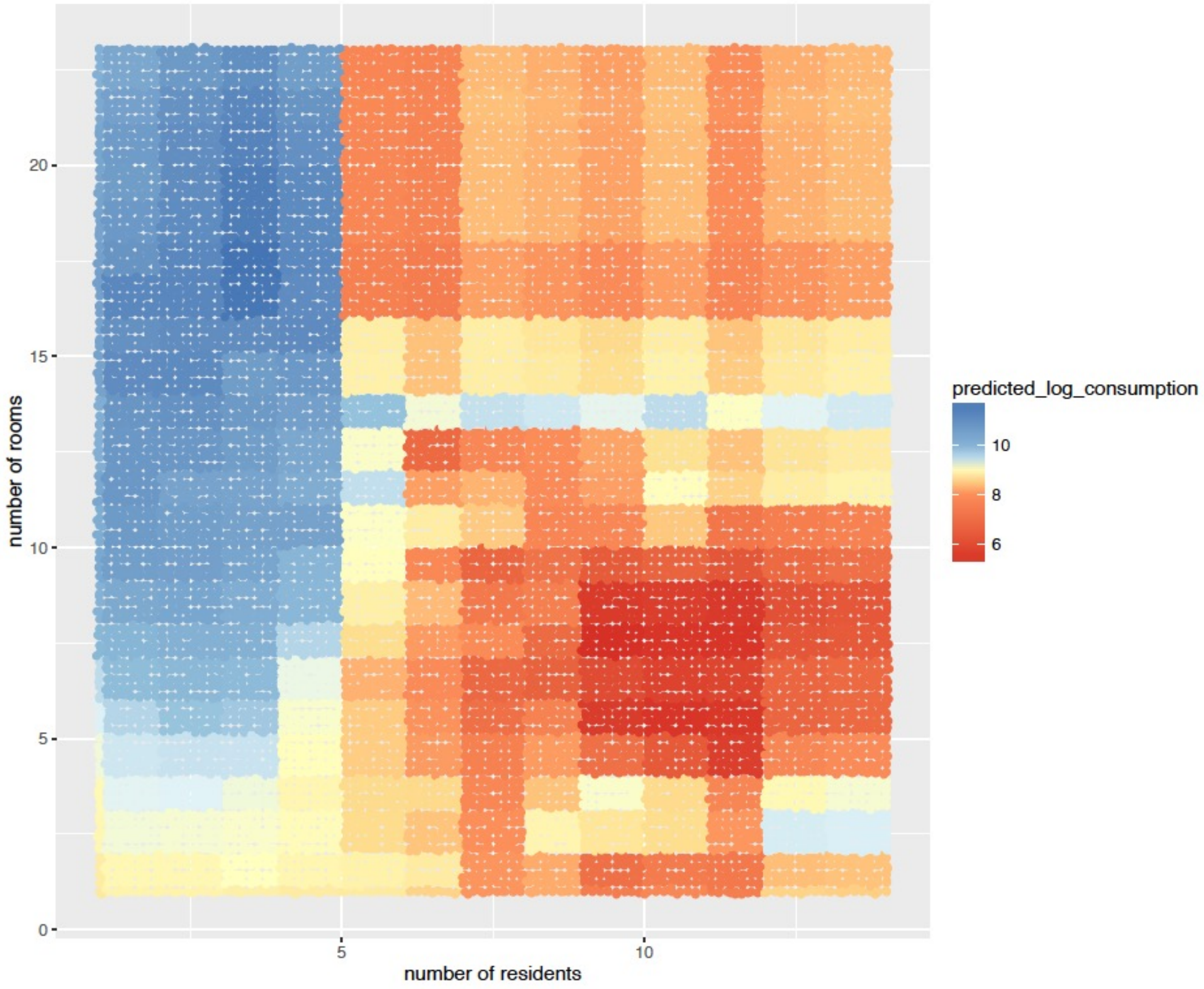
- Can combine across different model classes
- How to choose weights?

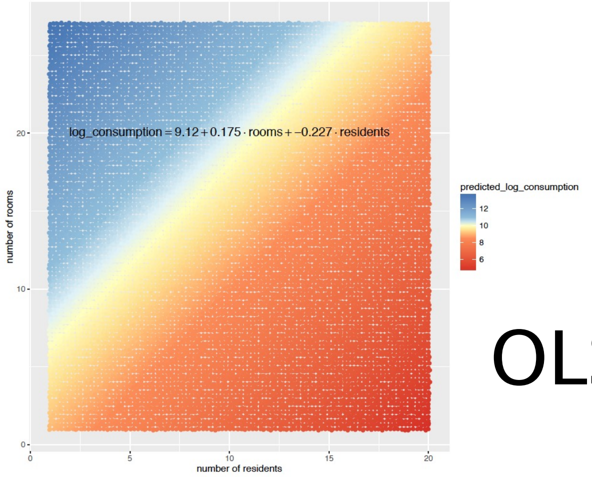
Model combination: bagging / random forest



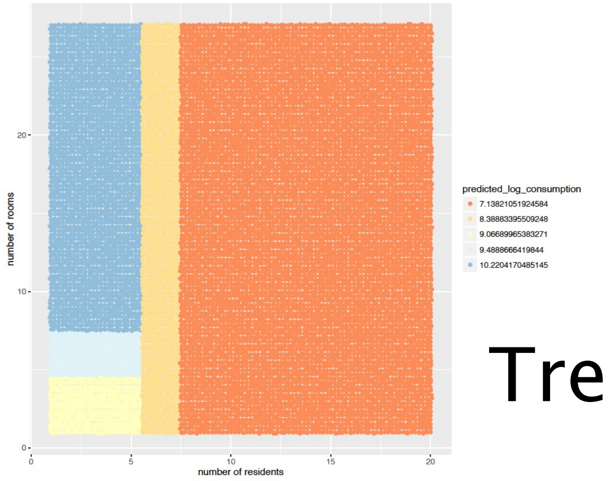


Random forest



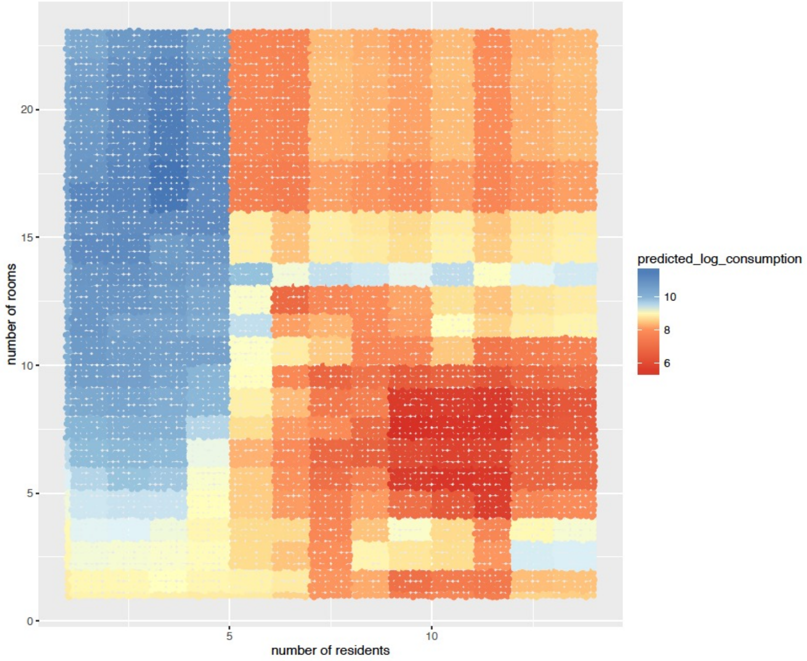


OLS



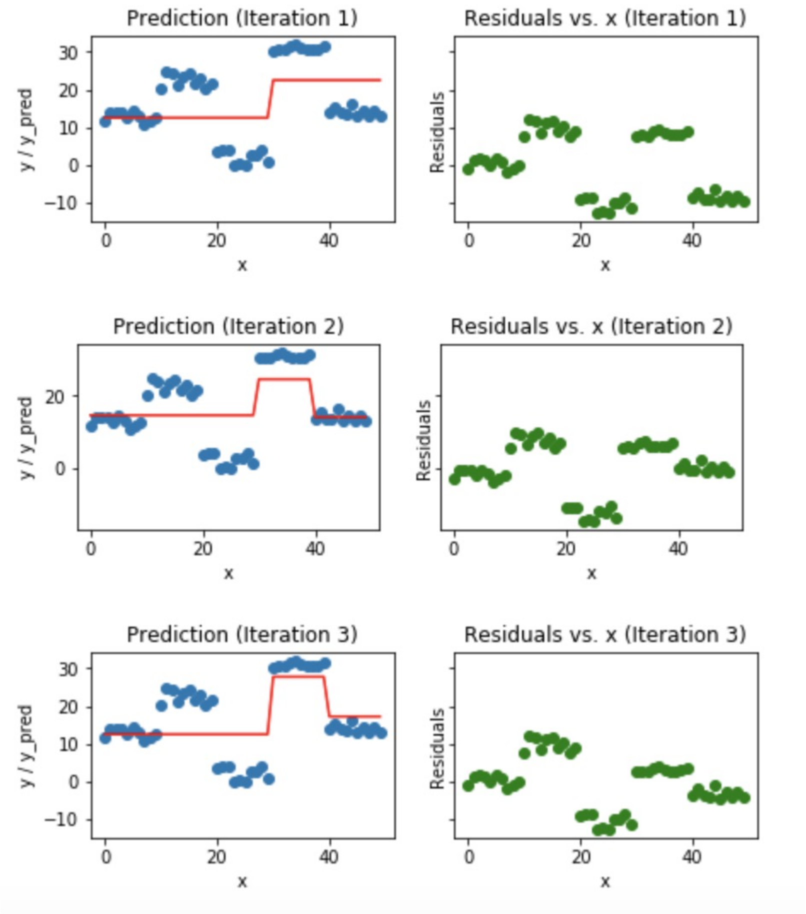
Tree

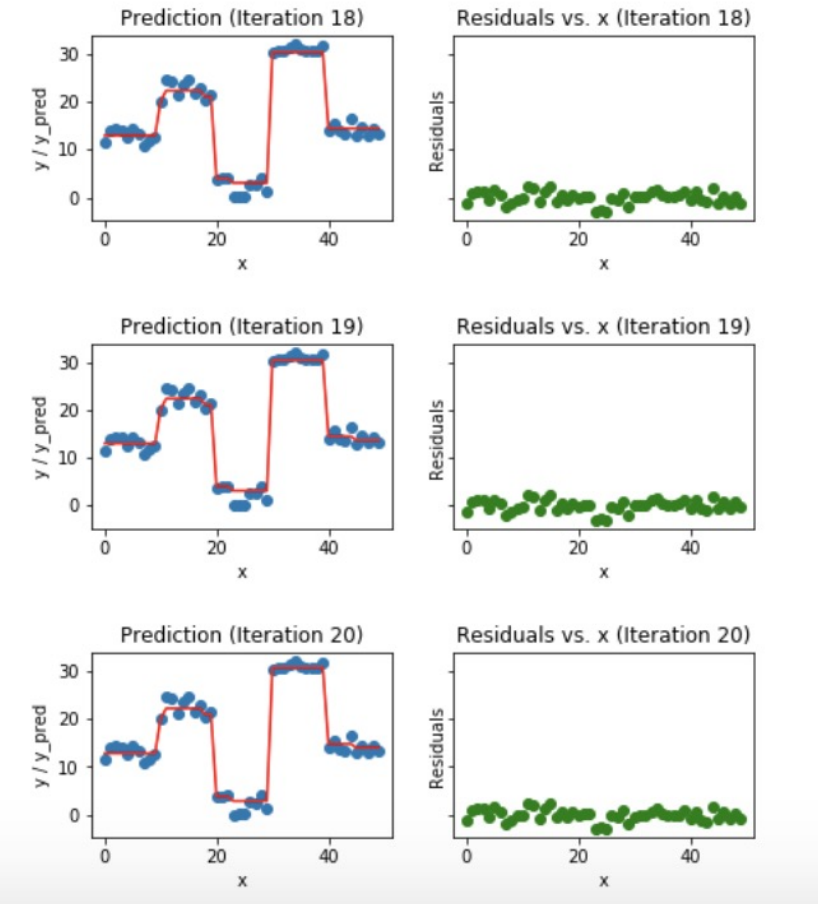
Forest



Boosting / boosted trees

- Iteratively fit a simple tree





Bayesian regularization

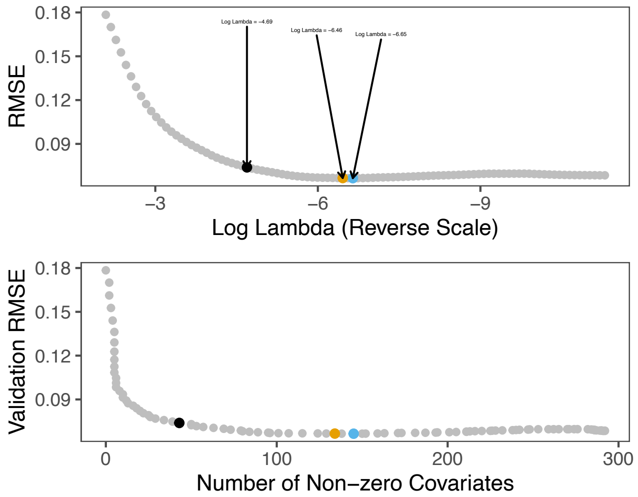
- Bayesian methods shrink towards a prior
- Powerful way of constructing regularized predictions, e.g. ridge regression, Bayesian trees

ML basics recap

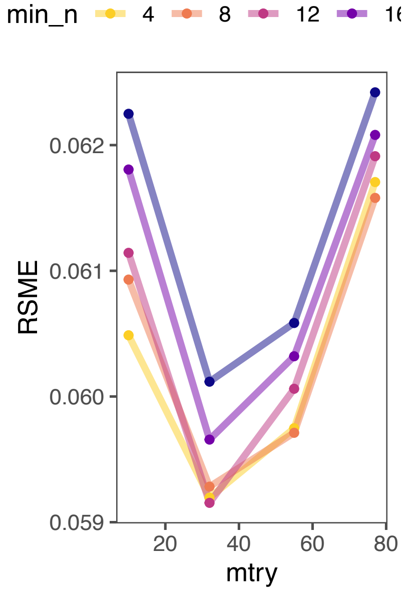
1. Flexible functional forms
2. Limit expressiveness (regularization)
3. Learn how much to regularize (tuning)

- Important researcher choices:
  - Loss function
  - Data management/splitting
  - Feature representation
  - Function class and regularizer

Implementation: R




```r
cv_lasso_fit <- cv.glmnet(x = XVars,
                          y = house_train$Sale_Price)
```




```r
cv_folds <- vfold_cv(house_train, v = 5)

rf_grid <- grid_regular(
  mtry(range = c(10, 100)),
  min_n(range = c(4, 20)),
  levels = 5
)
tune_rf_res <- tune_grid(
  tune_wf,
  resamples = cv_folds,
  grid = rf_grid
)
```

So what is new?

Statistics and econometrics

- Dominance of regularization: James and Stein (1961)
- Random forests: Breiman (2001)
- Non- and semiparametrics, sieve estimation

But still, something has happened

- Data
- Computation
- Functional forms that work
- Prediction focus that turns it into engineering competition
- Some new theoretical insights and developments, e.g. double descent, deep learning

ML basics recap

1. Flexible functional forms
2. Limit expressiveness (regularization)
3. Learn how much to regularize (tuning)

- What do these features imply for the properties of $\hat{f}$?
- And how can we therefore use $\hat{f}$ in applied work?

Structure of first chapter of webinar

1. Introduction

$$\text{Training data } (y, x) \xrightarrow{\hat{f}} \text{Application data } (\hat{y} = \hat{f}(x), x)$$

2. The Secret Sauce of Machine Learning

3. Prediction vs Estimation