CSE/STAT 416

Regularization – LASSO Regression

Amal Nanavati
University of Washington
July 6, 2022

Adapted from Hunter Schafer's Slides

Administrivia

Last lecture in the “Regression” case study!

- Next 2 weeks: Classification
- Following 1 week: Deep Learning

Section Tomorrow:

- Coding up RIDGE and Lasso (helpful for HW2!)

Upcoming Deadlines:

- HW1 Late deadline Thurs 7/7 11:59PM (if using 2 late days)
- HW2 out now, due Tues 7/12 11:59PM
- Learning Reflection 3 due Fri 11:59PM

Please monitor your grades on Canvas and reach out ASAP if there are any discrepancies. Canvas is the “final” gradebook.

OH is a great place to ask your learning reflection questions!

Reminder that we have lecture notes! Particularly recommended pre-lecture.

Poll on Brain Breaks.

Poll Everywhere

Think

30 sec

pollev.com/cs416

Would you prefer one ~10 min brain break, or two ~5 min brain breaks?

HW2 Walkthrough

Recap Ridge Regression & Address LR Uncertainties

What is min / argmin?

$$\min_x f(x)$$

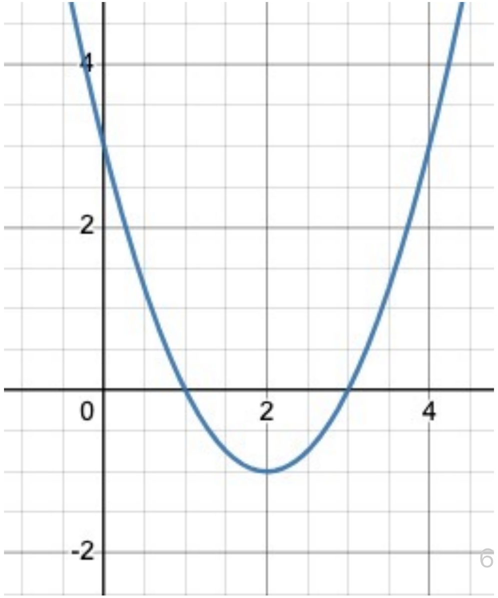
returns the minimum value of f(x)

$$\underset{x}{\operatorname{argmin}} f(x)$$

returns the value of x
for which $f(x)$ attains its minimum

Several Lec3 slides had min instead of argmin. I've fixed it on the website

$f(x) = x^2 - 4x + 3$



This should be randomized!

Cross-Validation

Clever idea: Use many small validation sets without losing too much training data.

Still need to break off our test set like before. After doing so, break the training set into k chunks.

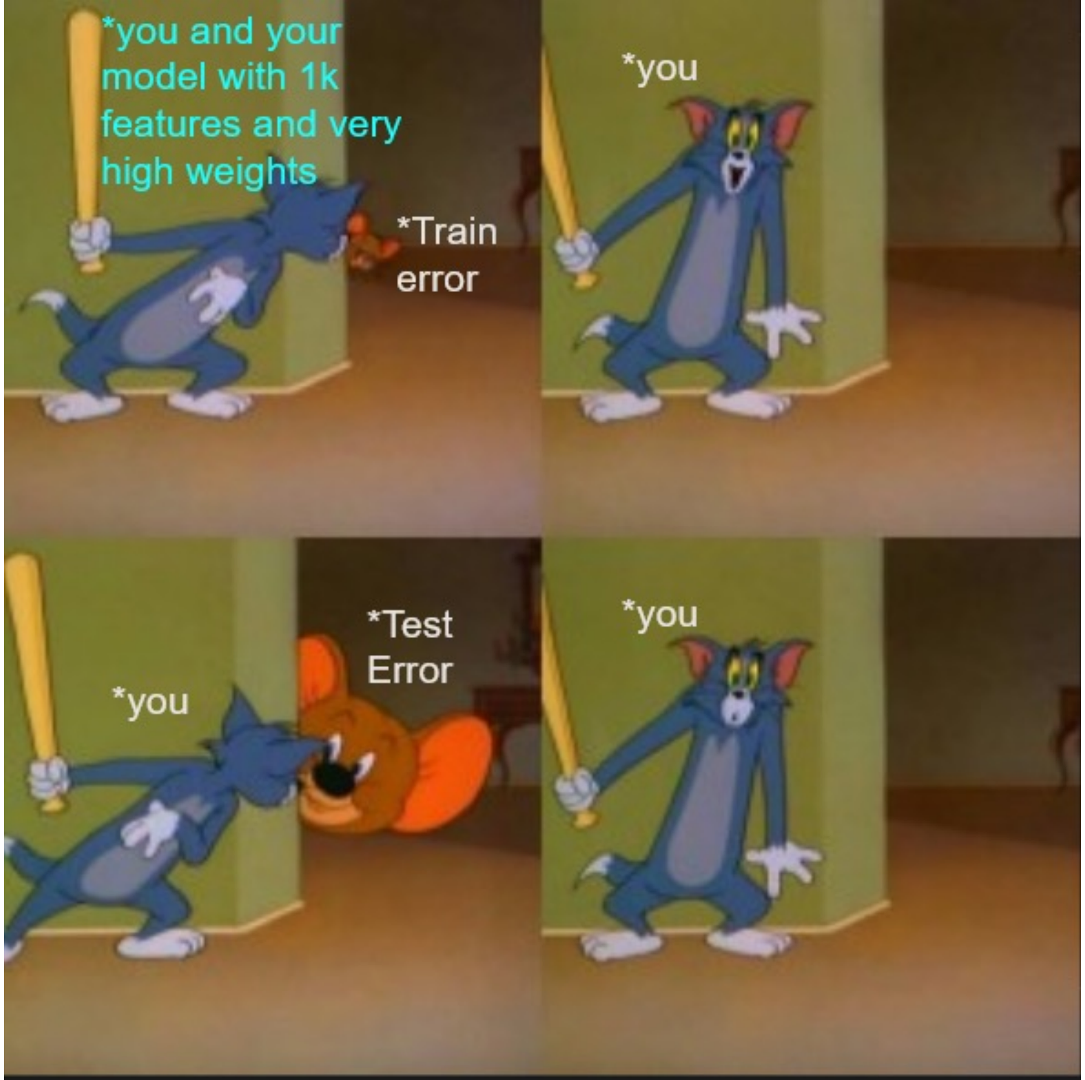
Train	Test

k chunks

Chunk1	Chunk2	Chunk3	Chunk4	Test

For a given model complexity, train it k times. Each time use all but one chunk and use that left out chunk to determine the validation error.

Overfitting In a Nutshell

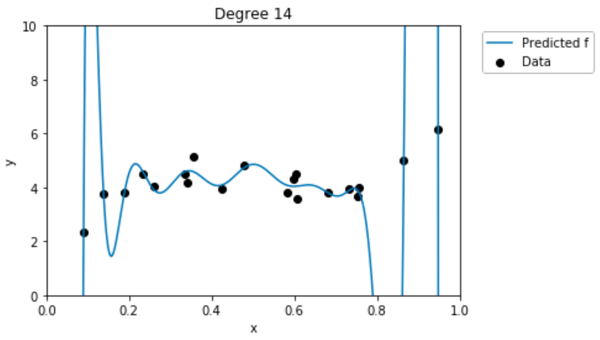


Number of Features

Overfitting is not limited to polynomial regression of large degree. It can also happen if you use a large number of features!

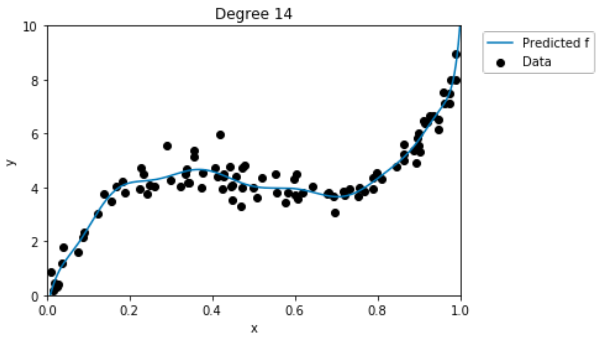
Why? Overfitting depends on whether the amount of data you have is large enough to represent the true function's complexity.

large $|\hat{w}_j|$



$\approx$ 20 pts

moderate $|\hat{w}_j|$



$\approx$100 pts

Ridge Regression

$$\hat{\omega} = \underset{\omega}{\text{argmin}}\ MSE(\omega) + \lambda \|\omega\|_2^2$$

Change quality metric to minimize

$$\hat{w} = \underset{w}{\text{argmin}}\ MSE(W) + \lambda \|w\|_2^2$$

λ is a tuning **hyperparameter** that changes how much the model cares about the regularization term.

What if $\lambda = 0$?

$$\hat{\omega} = \underset{\omega}{\text{argmin}}\ MSE(\omega)$$

$$= \hat{\omega}_{OLS}$$

Ordinary Least Square (OLS)

What if $\lambda = \infty$?

essentially : $\hat{\omega} = \underset{\omega}{\text{argmin}}\ \lambda \|\omega\|_2^2 = \vec{0}$

λ in between? $0 \leq \|\hat{\omega}_{ridge}\|_2 \leq \|\hat{\omega}_{OLS}\|_2$

Poll Everywhere

Think

~~3~~ Minutes
2

How does λ affect the bias and variance of the model? For each underlined section, select “Low” or “High” appropriately.

When $\lambda = 0$ $\Rightarrow$ Complex

The model has **(Low / High)** Bias and **(Low / High)** Variance.

When $\lambda = \infty$ $\Rightarrow$ Simple

The model has **(Low / High)** Bias and **(Low / High)** Variance.

Choosing λ

The process for selecting λ is exactly the same as we saw with using a validation set or using cross validation.

for λ in λs:

Train a model using using Gradient Descent

$$\hat{w}_{ridge(\lambda)} = \underset{w}{\operatorname{argmin}} MSE_{train}(w) + \lambda||w_{1:D}||_2^2$$

Compute validation error

$$validation\_error = MSE_{val}(\hat{w}_{ridge(\lambda)})$$

Track λ with smallest $validation\_error$

Return λ^* & estimated future error $MSE_{test}(\hat{w}_{ridge(\lambda^*)})$

Scaling Features

Fix this by **normalizing** the features so all are on the same scale!

$$\tilde{h}_j(x_i) = \frac{h_j(x_i) - \mu_j(x_1, \dots, x_N)}{\sigma_j(x_1, \dots, x_N)}$$

For feature j:

μ_j mean

σ_j std. dev.

$$\boxed{\tilde{h}_j(x) = \frac{h_j(x) - \mu_j}{\sigma_j}}$$

Where

The mean of feature j:

$$\mu_j(x_1, \dots, x_N) = \frac{1}{N}\sum_{i=1}^{N} h_j(x_i)$$

The standard devation of feature j:

$$\sigma_j(x_1, \dots, x_N) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(h_j(x_i) - \mu_j(x_1, \dots, x_N)\right)^2}$$

Important: Must scale the test data and all future data using the means and standard deviations **of the training set!**

Otherwise the units of the model and the units of the data are not comparable!

Poll Everywhere

Think

1 min

What are some real-world / human analogies for each of these concepts?

- Overfitting / Underfitting
- Train Set / Test Set
- Bias
- Variance
- Regularization

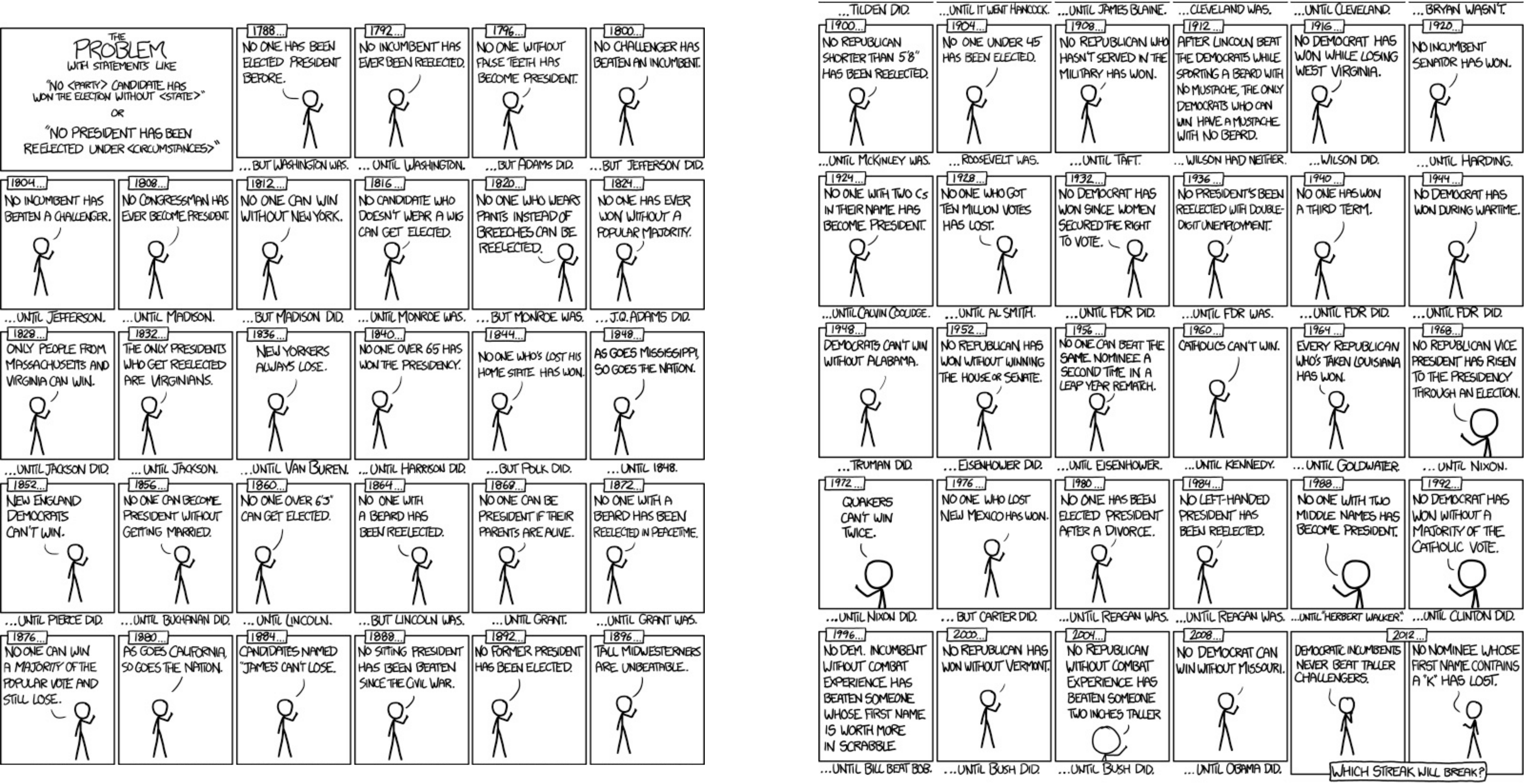
Poll Everywhere

Group

2 mins

What are some real-world / human analogies for each of these concepts?

- Overfitting / Underfitting
- Train Set / Test Set
- Bias
- Variance
- Regularization


THE PROBLEM WITH STATEMENTS LIKE
"NO <PARTY> CANDIDATE HAS WON THE ELECTION WITHOUT <STATE>"
OR
"NO PRESIDENT HAS BEEN REELECTED UNDER <CIRCUMSTANCES>"
1788...
NO ONE HAS BEEN ELECTED PRESIDENT BEFORE.
...BUT WASHINGTON WAS.
1792...
NO INCUMBENT HAS EVER BEEN REELECTED.
...UNTIL WASHINGTON.
1796...
NO ONE WITHOUT FALSE TEETH HAS BECOME PRESIDENT.
...BUT ADAMS DID.
1800...
NO CHALLENGER HAS BEATEN AN INCUMBENT.
...BUT JEFFERSON DID.
1804...
NO INCUMBENT HAS BEATEN A CHALLENGER.
...UNTIL JEFFERSON.
1808...
NO CONGRESSMAN HAS EVER BECOME PRESIDENT.
...UNTIL MADISON.
1812...
NO ONE CAN WIN WITHOUT NEW YORK.
...BUT MADISON DID.
1816...
NO CANDIDATE WHO DOESN'T WEAR A WIG CAN GET ELECTED.
...UNTIL MONROE WAS.
1820...
NO ONE WHO WEARS PANTS INSTEAD OF BREECHES CAN BE REELECTED.
...BUT MONROE WAS.
1824...
NO ONE HAS EVER WON WITHOUT A POPULAR MAJORITY.
...J.Q. ADAMS DID.
1828...
ONLY PEOPLE FROM MASSACHUSETTS AND VIRGINIA CAN WIN.
...UNTIL JACKSON DID.
1832...
THE ONLY PRESIDENTS WHO GET REELECTED ARE VIRGINIANS.
...UNTIL JACKSON.
1836...
NEW YORKERS ALWAYS LOSE.
...UNTIL VAN BUREN.
1840...
NO ONE OVER 65 HAS WON THE PRESIDENCY.
...UNTIL HARRISON DID.
1844...
NO ONE WHO'S LOST HIS HOME STATE HAS WON.
...BUT POLK DID.
1848...
AS GOES MISSISSIPPI, SO GOES THE NATION.
...UNTIL 1848.
1852...
NEW ENGLAND DEMOCRATS CAN'T WIN.
...UNTIL PIERCE DID.
1856...
NO ONE CAN BECOME PRESIDENT WITHOUT GETTING MARRIED.
...UNTIL BUCHANAN DID.
1860...
NO ONE OVER 6'3" CAN GET ELECTED.
...UNTIL LINCOLN.
1864...
NO ONE WITH A BEARD HAS BEEN REELECTED.
...BUT LINCOLN WAS.
1868...
NO ONE CAN BE PRESIDENT IF THEIR PARENTS ARE ALIVE.
...UNTIL GRANT.
1872...
NO ONE WITH A BEARD HAS BEEN REELECTED IN PEACETIME.
...UNTIL GRANT WAS.
1876...
NO ONE CAN WIN A MAJORITY OF THE POPULAR VOTE AND STILL LOSE.
...TILDEN DID.
1880...
AS GOES CALIFORNIA, SO GOES THE NATION.
...UNTIL IT WENT HANCOCK.
1884...
CANDIDATES NAMED "JAMES" CAN'T LOSE.
...UNTIL JAMES BLAINE.
1888...
NO SITTING PRESIDENT HAS BEEN BEATEN SINCE THE CIVIL WAR.
...CLEVELAND WAS.
1892...
NO FORMER PRESIDENT HAS BEEN ELECTED.
...UNTIL CLEVELAND.
1896...
TALL MIDWESTERNERS ARE UNBEATABLE.
...BRYAN WASN'T.
1900...
NO REPUBLICAN SHORTER THAN 5'8" HAS BEEN REELECTED.
...UNTIL McKINLEY WAS.
1904...
NO ONE UNDER 45 HAS BEEN ELECTED.
...ROOSEVELT WAS.
1908...
NO REPUBLICAN WHO HASN'T SERVED IN THE MILITARY HAS WON.
...UNTIL TAFT.
1912...
AFTER LINCOLN BEAT THE DEMOCRATS WHILE SPORTING A BEARD WITH NO MUSTACHE, THE ONLY DEMOCRATS WHO CAN WIN HAVE A MUSTACHE WITH NO BEARD.
...WILSON HAD NEITHER.
1916...
NO DEMOCRAT HAS WON WHILE LOSING WEST VIRGINIA.
...WILSON DID.
1920...
NO INCUMBENT SENATOR HAS WON.
...UNTIL HARDING.
1924...
NO ONE WITH TWO Cs IN THEIR NAME HAS BECOME PRESIDENT.
...UNTIL CALVIN COOLIDGE.
1928...
NO ONE WHO GOT TEN MILLION VOTES HAS LOST.
...UNTIL AL SMITH.
1932...
NO DEMOCRAT HAS WON SINCE WOMEN SECURED THE RIGHT TO VOTE.
...UNTIL FDR DID.
1936...
NO PRESIDENT'S BEEN REELECTED WITH DOUBLE-DIGIT UNEMPLOYMENT.
...UNTIL FDR WAS.
1940...
NO ONE HAS WON A THIRD TERM.
...UNTIL FDR DID.
1944...
NO DEMOCRAT HAS WON DURING WARTIME.
...UNTIL FDR DID.
1948...
DEMOCRATS CAN'T WIN WITHOUT ALABAMA.
...TRUMAN DID.
1952...
NO REPUBLICAN HAS WON WITHOUT WINNING THE HOUSE OR SENATE.
...EISENHOWER DID.
1956...
NO ONE CAN BEAT THE SAME NOMINEE A SECOND TIME IN A LEAP YEAR REMATCH.
...UNTIL EISENHOWER.
1960...
CATHOLICS CAN'T WIN.
...UNTIL KENNEDY.
1964...
EVERY REPUBLICAN WHO'S TAKEN LOUISIANA HAS WON.
...UNTIL GOLDWATER.
1968...
NO REPUBLICAN VICE PRESIDENT HAS RISEN TO THE PRESIDENCY THROUGH AN ELECTION.
...UNTIL NIXON.
1972...
QUAKERS CAN'T WIN TWICE.
...UNTIL NIXON DID.
1976...
NO ONE WHO LOST NEW MEXICO HAS WON.
...BUT CARTER DID.
1980...
NO ONE HAS BEEN ELECTED PRESIDENT AFTER A DIVORCE.
...UNTIL REAGAN WAS.
1984...
NO LEFT-HANDED PRESIDENT HAS BEEN REELECTED.
...UNTIL REAGAN WAS.
1988...
NO ONE WITH TWO MIDDLE NAMES HAS BECOME PRESIDENT.
...UNTIL "HERBERT WALKER."
1992...
NO DEMOCRAT HAS WON WITHOUT A MAJORITY OF THE CATHOLIC VOTE.
...UNTIL CLINTON DID.
1996...
NO DEM. INCUMBENT WITHOUT COMBAT EXPERIENCE HAS BEATEN SOMEONE WHOSE FIRST NAME IS WORTH MORE IN SCRABBLE.
...UNTIL BILL BEAT BOB.
2000...
NO REPUBLICAN HAS WON WITHOUT VERMONT.
...UNTIL BUSH DID.
2004...
NO REPUBLICAN WITHOUT COMBAT EXPERIENCE HAS BEATEN SOMEONE TWO INCHES TALLER.
...UNTIL BUSH DID.
2008...
NO DEMOCRAT CAN WIN WITHOUT MISSOURI.
...UNTIL OBAMA DID.
2012...
DEMOCRATIC INCUMBENTS NEVER BEAT TALLER CHALLENGERS.
NO NOMINEE WHOSE FIRST NAME CONTAINS A "K" HAS LOST.
WHICH STREAK WILL BREAK?

Feature Selection & All Subsets

Benefits

Why do we care about selecting features? Why not use them all?

Complexity

Models with too many features are more complex. Might overfit!

Interpretability

Can help us identify which features carry more information.

Efficiency

Imagine if we had MANY features (e.g. DNA). $\hat{w}$ could have 10^{11} coefficients. Evaluating $\hat{y} = \hat{w}^T h(x)$ would be very slow!

If $\hat{w}$ is **sparse**, only need to look at the non-zero coefficients

$$\hat{y} = \sum_{\hat{w}_j \neq 0} \hat{w}_j h_j(x)$$

Sparsity: Housing

Might have many features to potentially use. Which are useful?

Lot size
Single Family
Year built
Last sold price
Last sale price/sqft
Finished sqft
Unfinished sqft
Finished basement sqft
floors
Flooring types
Parking type
Parking amount
Cooling
Heating
Exterior materials
Roof type
Structure style

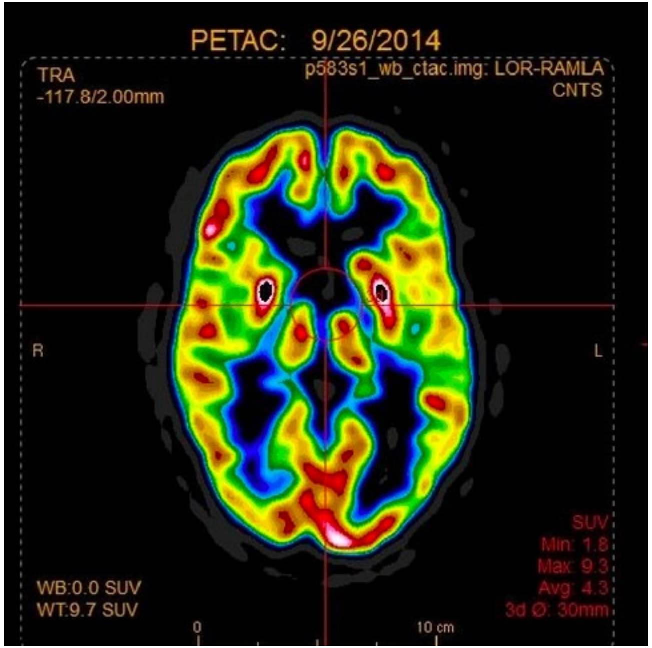
Dishwasher
Garbage disposal
Microwave
Range / Oven
Refrigerator
Washer
Dryer
Laundry location
Heating type
Jetted Tub
Deck
Fenced Yard
Lawn
Garden
Sprinkler System

…

Sparsity: Reading Minds

How happy are you? What part of the brain controls happiness?

x

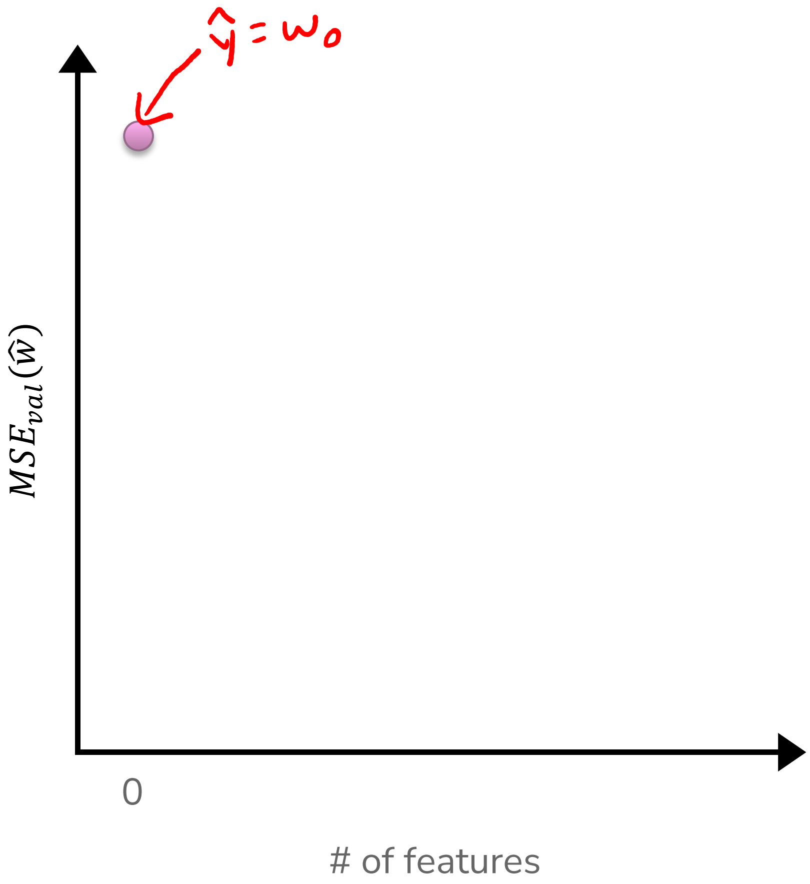


y

10 – Happy ☺

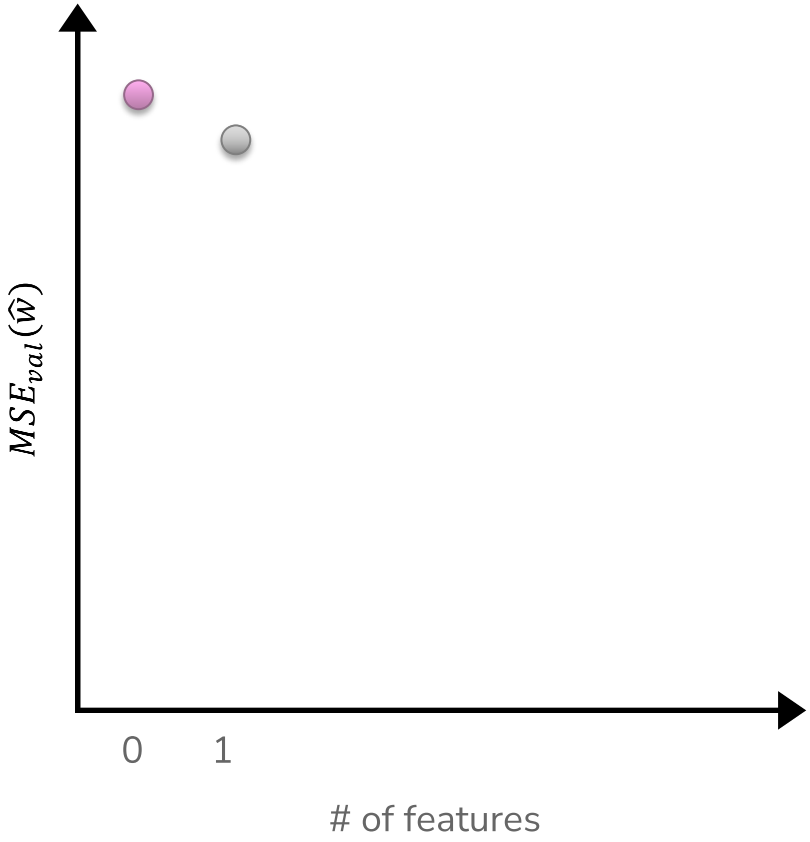
0 – Sad ☹

Best Model Size 0



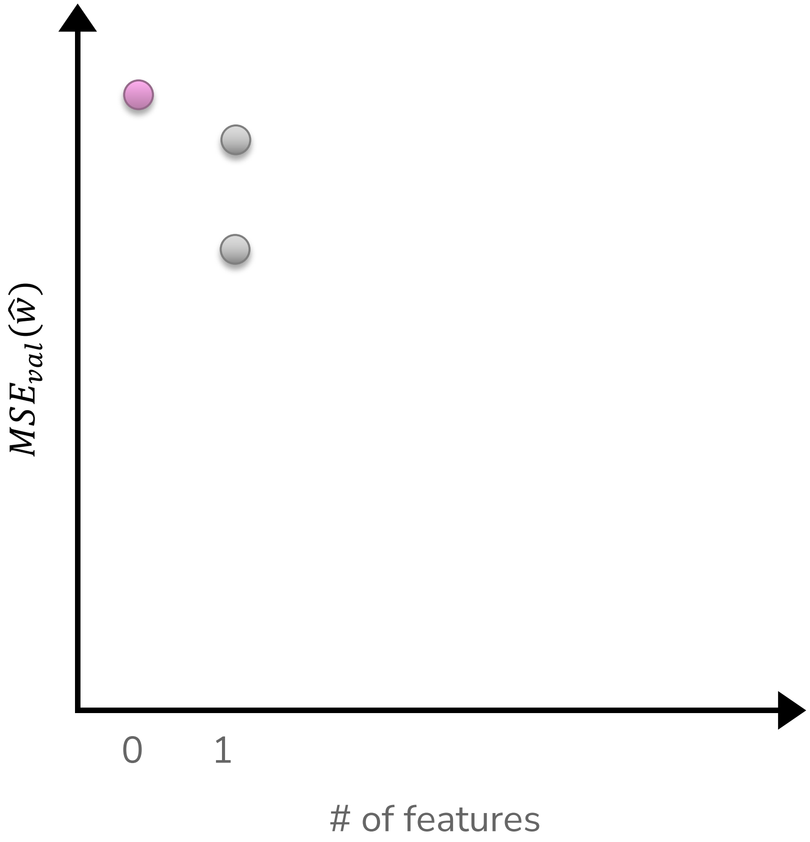
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



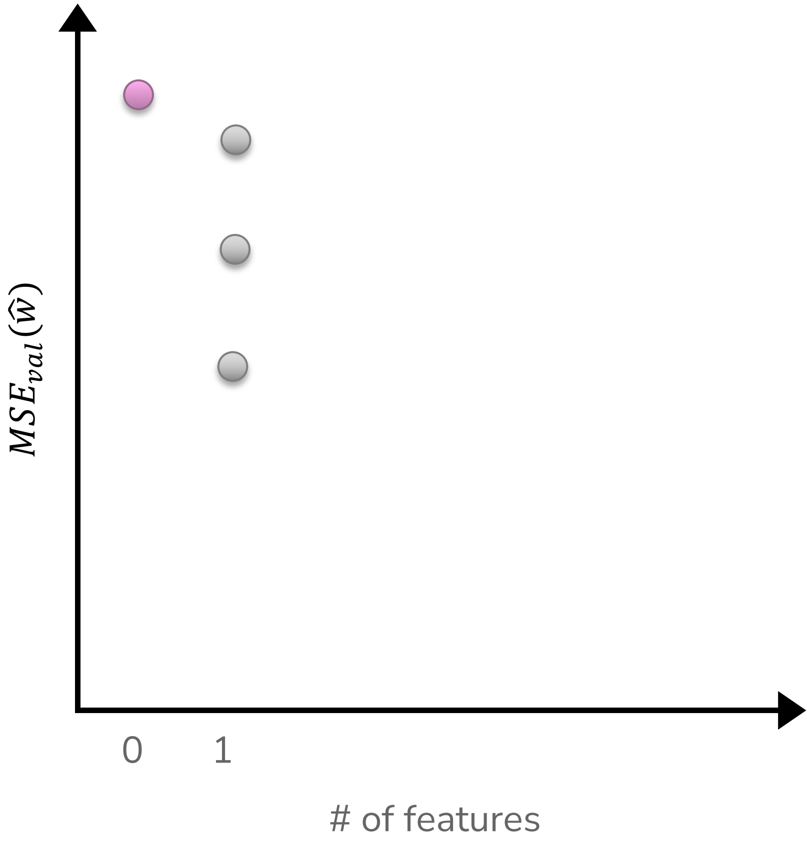
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



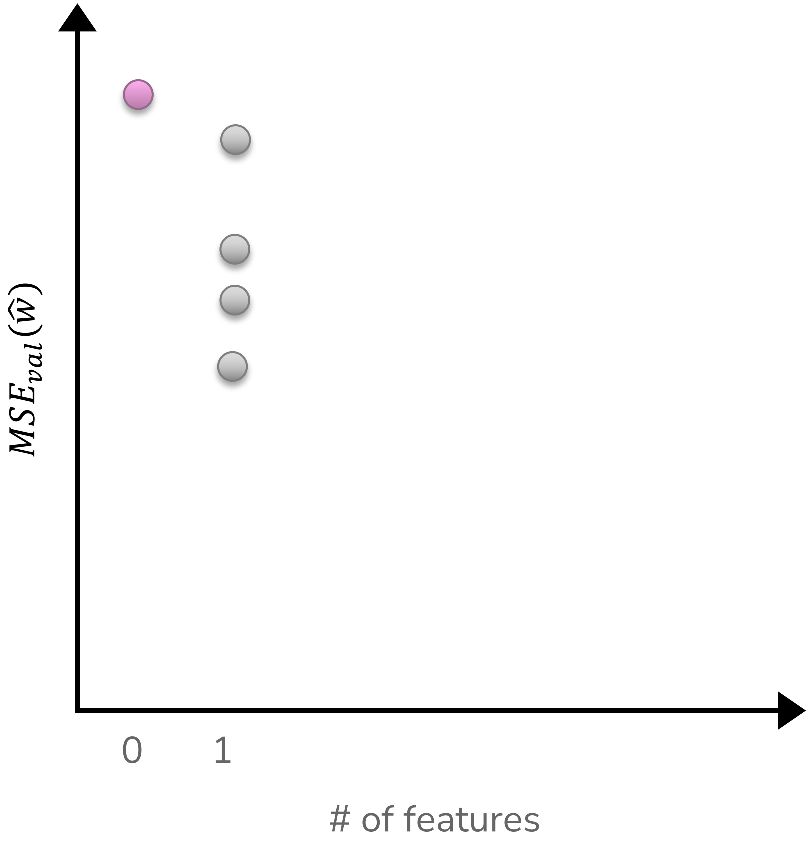
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



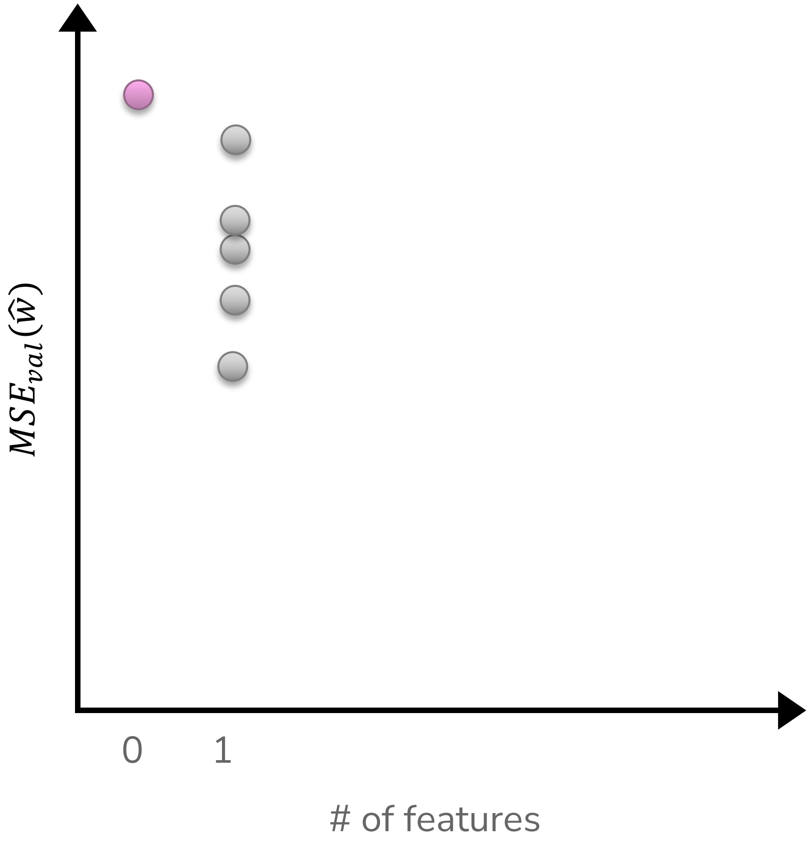
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



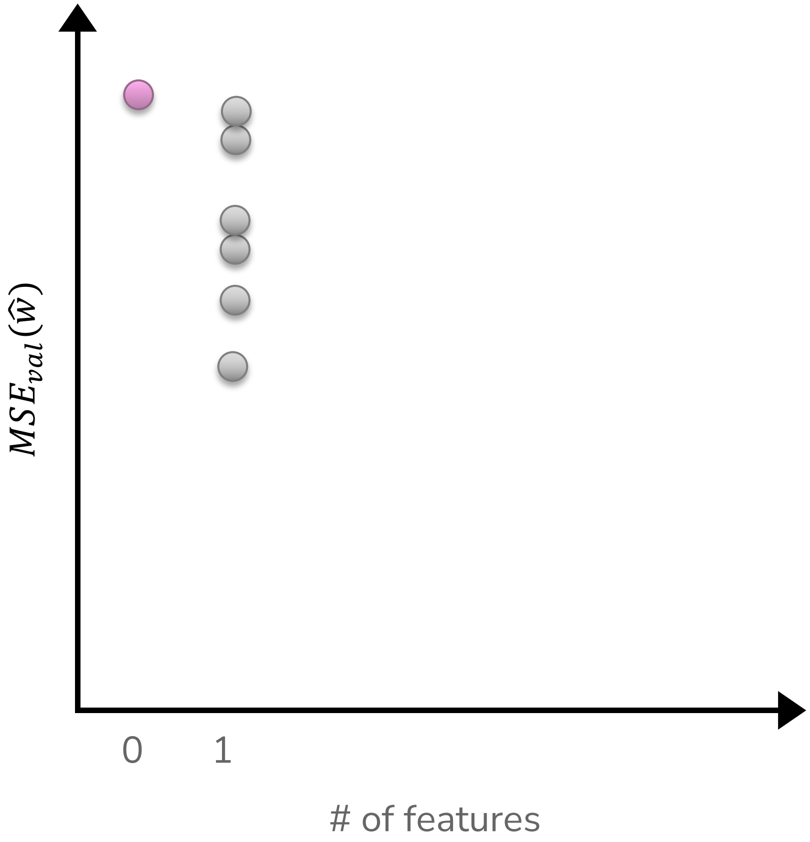
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



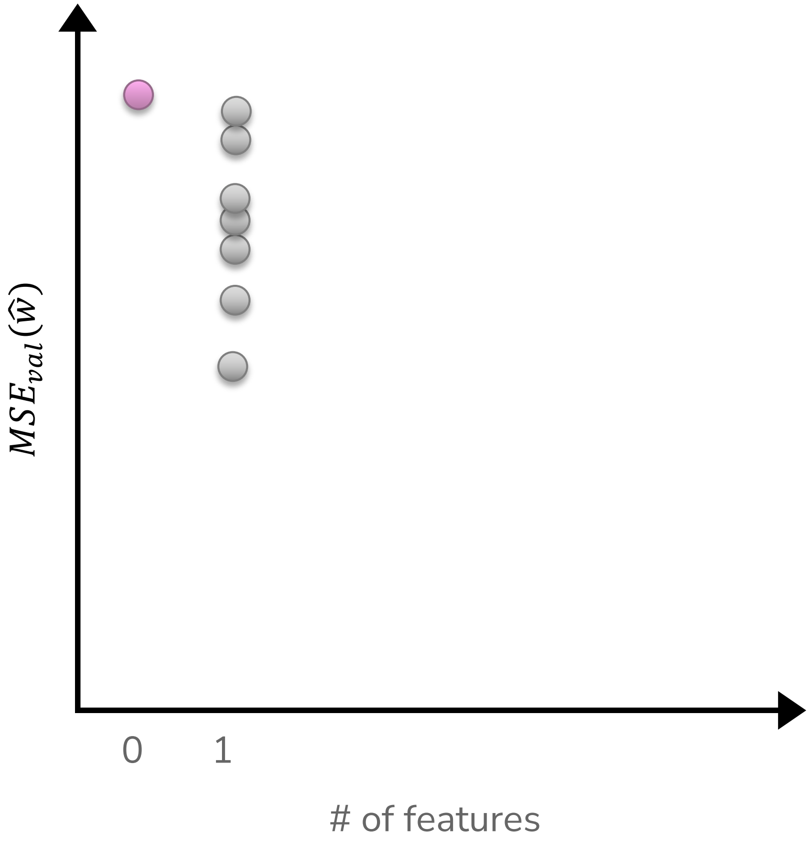
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



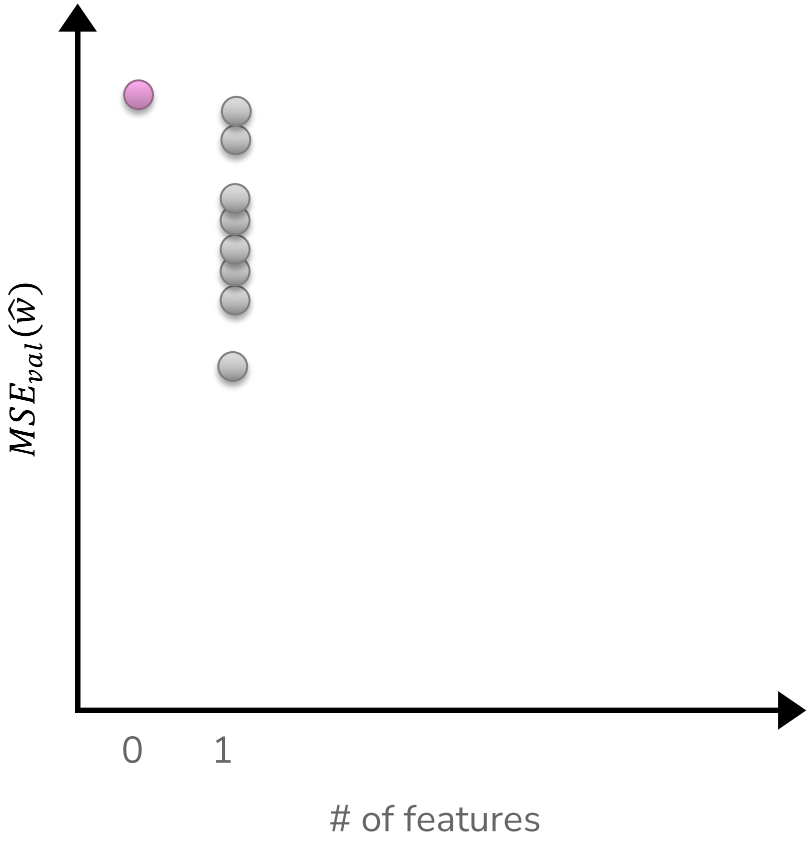
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



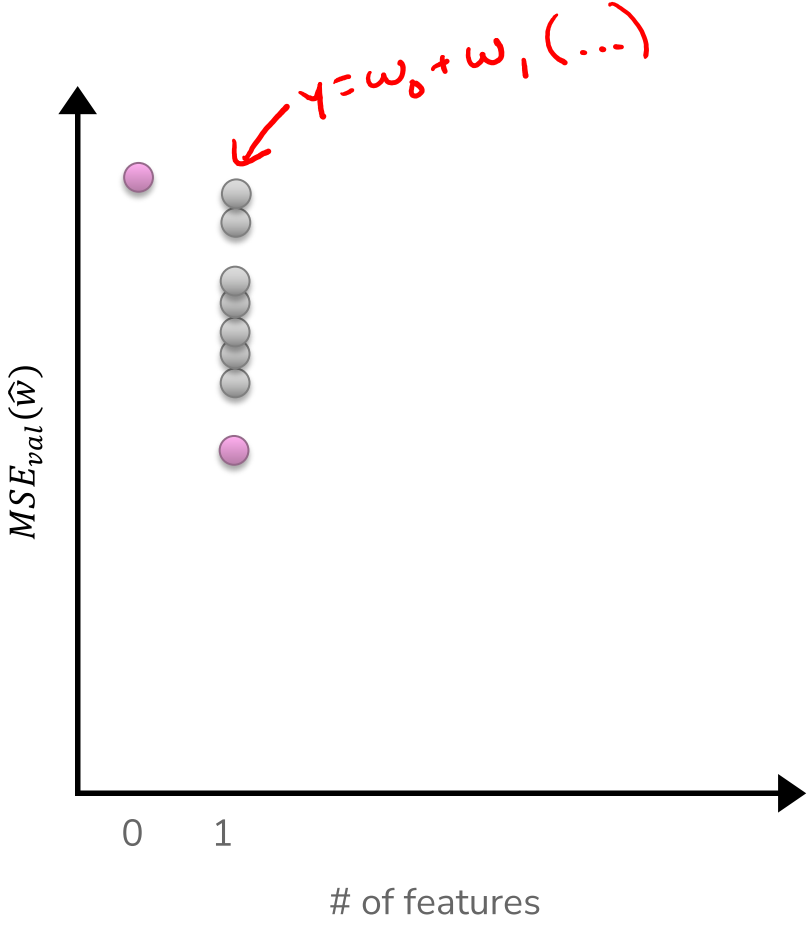
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 1



Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

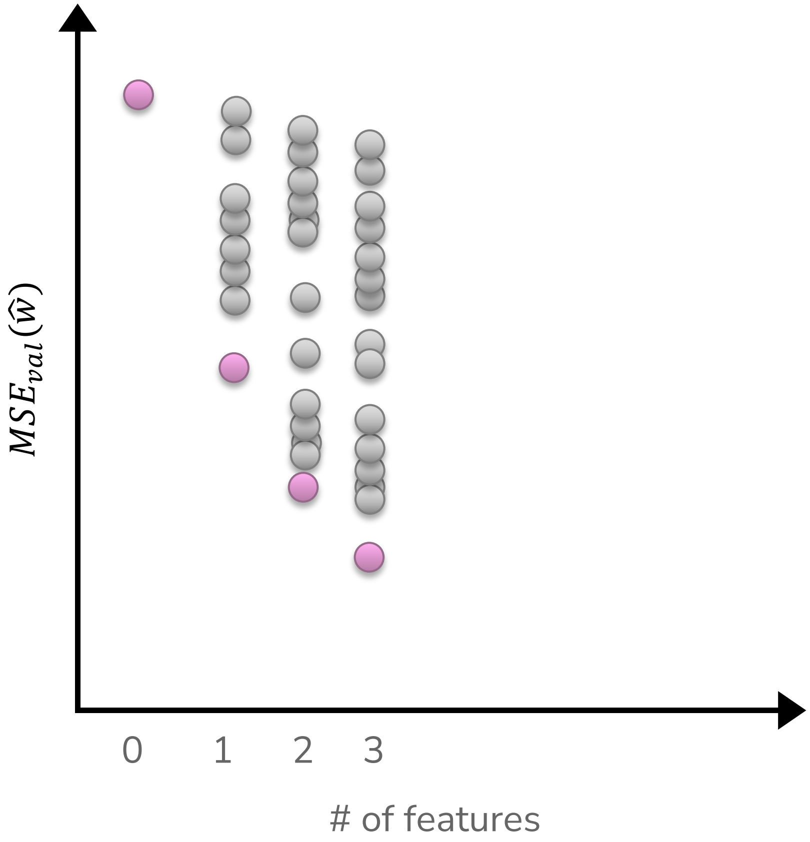
NOT NECESSARILY NESTED!!!

Best Model Size 2

Not necessarily nested!
Best Model – Size 1: sq.ft living
Best Model – Size 2:
bathrooms & # bedrooms

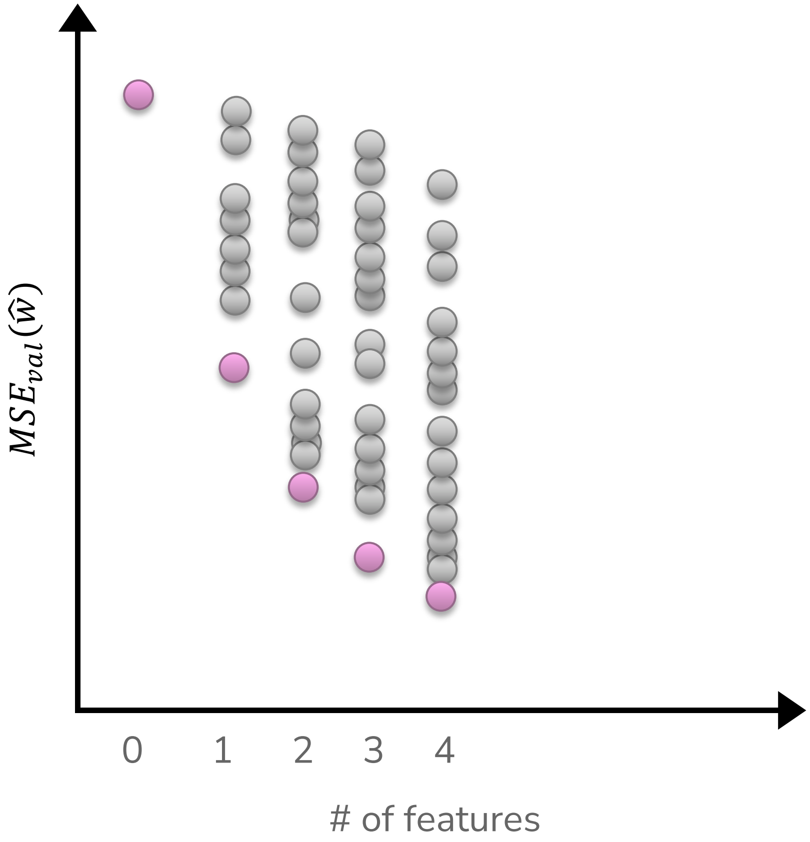
Features
- # bathrooms
- # bedrooms
- sq.ft. living
- sq.ft lot
- floors
- year built
- year renovated
- waterfront

Best Model Size 3



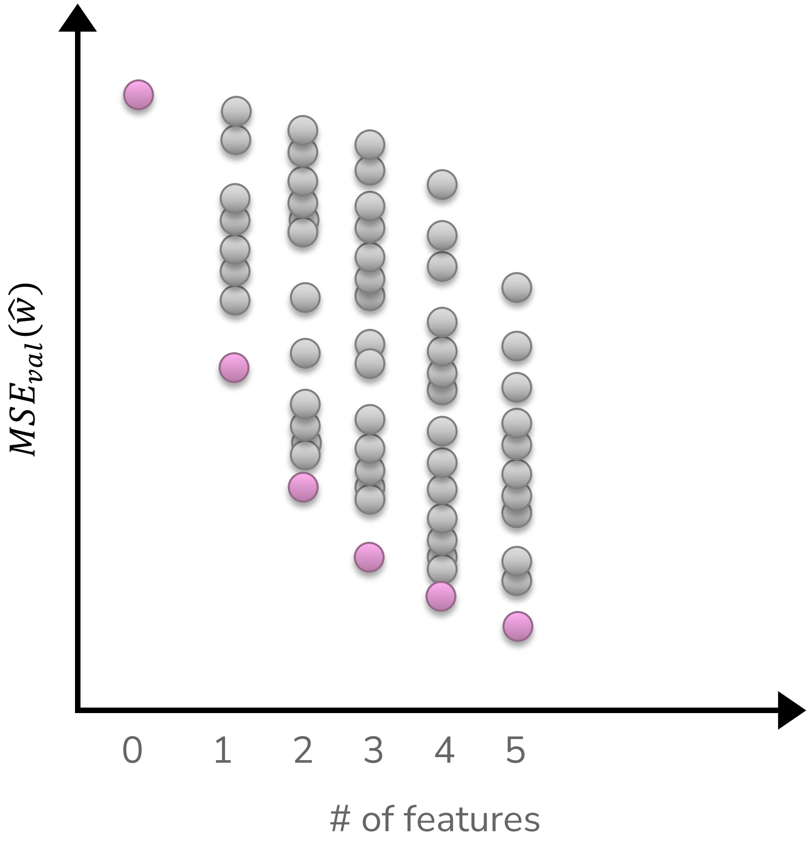
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 4



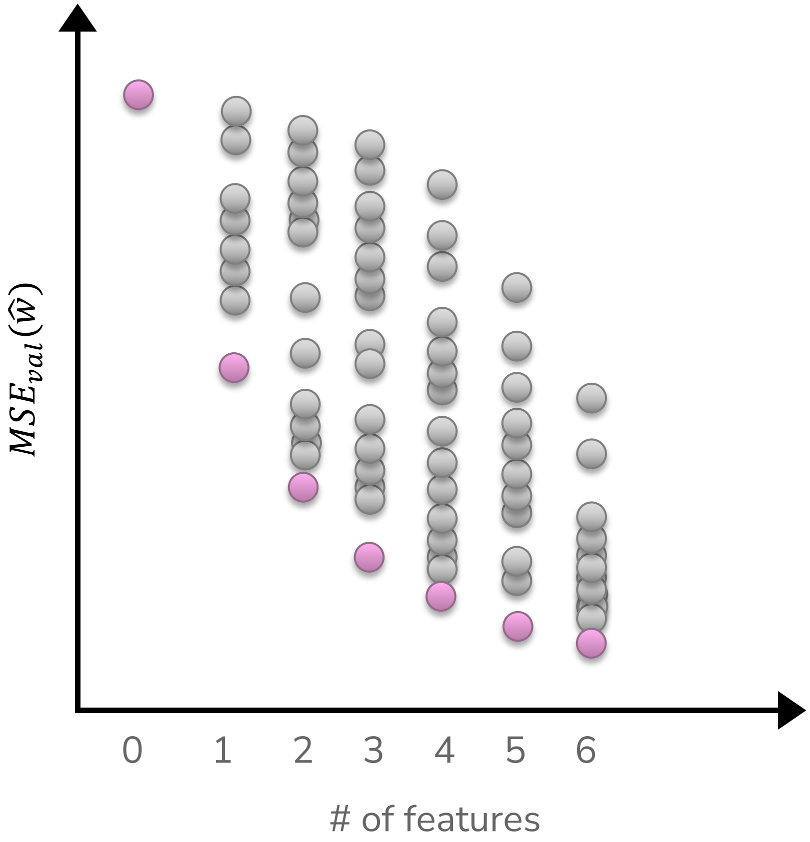
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 5



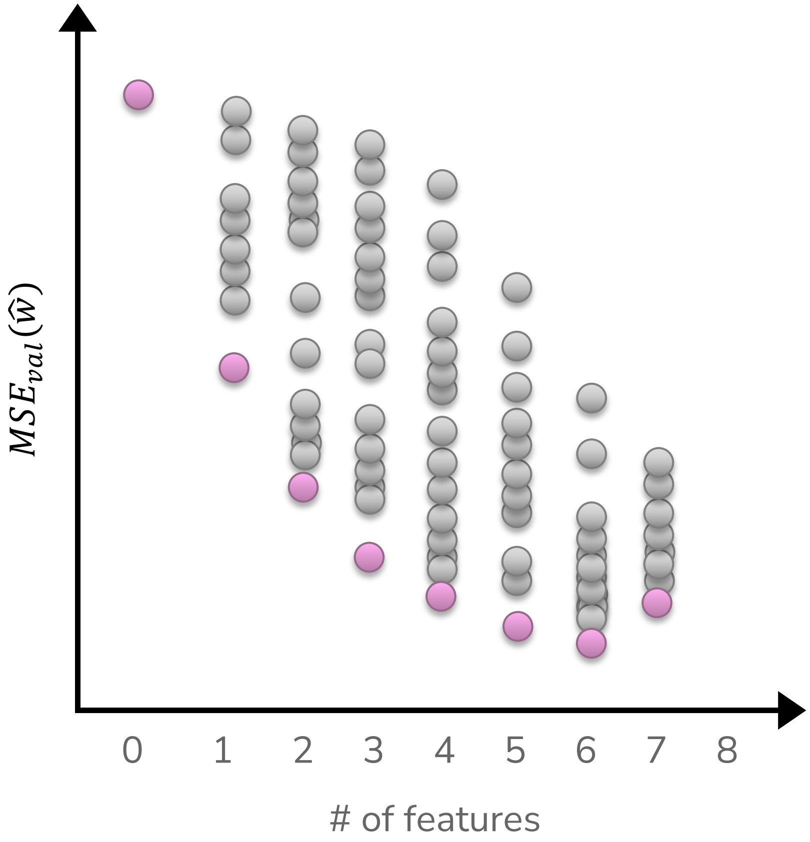
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 6



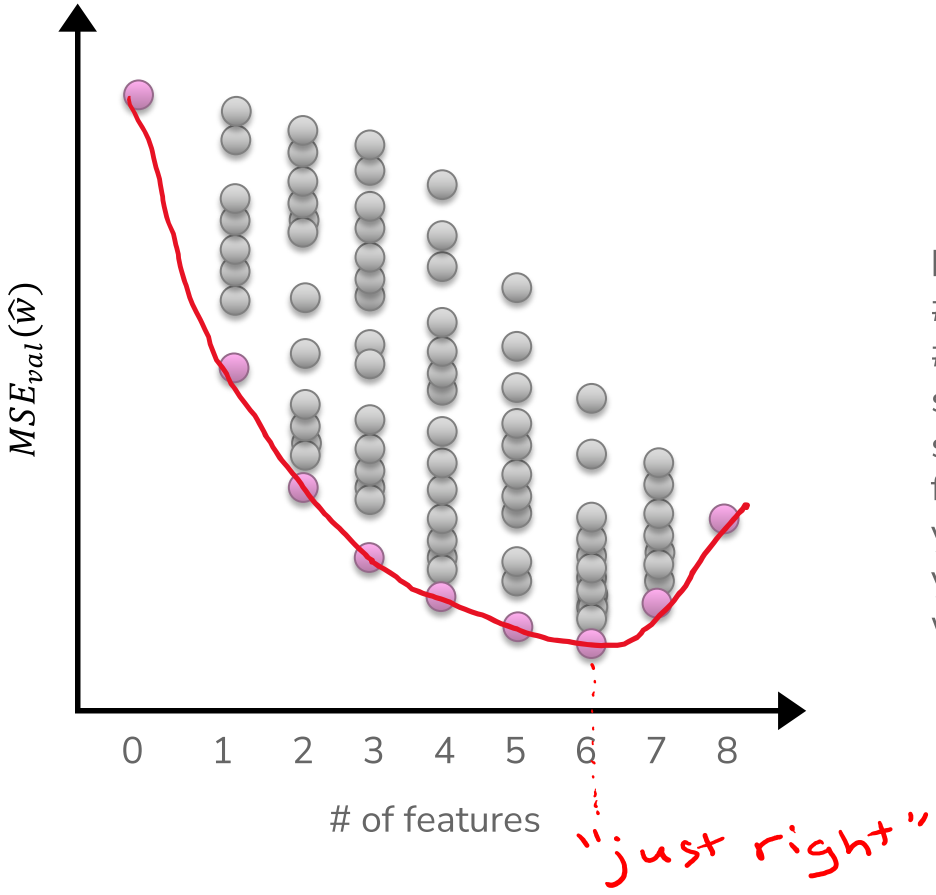
Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model Size 7



Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Best Model
Size 8



Features
bathrooms
bedrooms
sq.ft. living
sq.ft lot
floors
year built
year renovated
waterfront

Efficiency of All Subsets

How many models did we evaluate?

whether $h_1(x)$ is in the dataset

$h_D(x)$

$\hat{y}_i = w_0$	[0 0 0 … 0 0 0]
$\hat{y}_i = w_0 + w_1 h_1(x)$	[1 0 0 … 0 0 0]
$\hat{y}_i = w_0 + w_2 h_2(x)$	[0 1 0 … 0 0 0]
…	…
$\hat{y}_i = w_0 + w_1 h_1(x) + w_2 h_2(x)$	[1 1 0 … 0 0 0]
…	…
$\hat{y}_i = w_0 + w_1 h_1(x) + \ldots + w_D h_D(x)$	[1 1 1 … 1 1 1]

2^D possible combinations

If evaluating all subsets of 8 features only took 5 seconds, then

- 16 features would take 21 minutes
- 32 features would take almost 3 years
- 100 features would take almost 7.5×10^{20} years
  - 50,000,000,000x longer than the age of the universe!

Poll Everywhere

Think

1 min

pollev.com/cs416

We've seen that using a validation set to find the best polynomial degree (from 0 to $p-1$) requires training p models.

for deg in $[0, \ldots, p-1]$:

- train model w/ degree deg on train set
- get val error

return model w/ least val error

Say you have a dataset with d input columns, and you're using the all subset's approach to find the best features for your model. How many models would you train?

A. d

B. $d-1$

C. $2d$

D. 2^d

E. 2^{d+1}

Poll Everywhere

Group

2 min

pollev.com/cs416

We've seen that using a validation set to find the best polynomial degree (from 0 to $p-1$) requires training p models.

Say you have a dataset with d input columns, and you're using the all subset's approach to find the best features for your model. How many models would you train?

A. d

B. $d-1$

C. $2d$

✓ D. 2^d

E. 2^{d+1} ← can be this if we vary whether we have an intercept

Choose Num Features?

Clearly all subsets is unreasonable. How can we choose how many and which features to include?

Option 1

Greedy Algorithm

Option 2

LASSO Regression (L1 Regularization)

L2 ⇒ Ridge

L1 ⇒ LASSO

Brain Break

Greedy Algorithms

Greedy Algorithms

Knowing it's impossible to find exact solution, approximate it!

Greedy Algo takes locally optimally steps

Forward stepwise

Start from model with no features, iteratively add features as performance improves.

Backward stepwise

Start with a full model and iteratively remove features that are the least useful.

Combining forward and backwards steps

Do a forward greedy algorithm that eventually prunes features that are no longer as relevant

And many many more!

Forward Stepwise

(Example Greedy Algorithm)

Start by selecting number of features k

```
min_val = ∞
S_0 ← ∅
for i ← 1..k:
    Find feature f_i not in S_{i-1}, that when combined
    with S_{i-1}, minimizes the validation loss the most.
    S_i ← S_{i-1} ∪ {f_i}
    if val_loss(S_i) > min_val:
        break
```

selected feature

trying every feature remaining, one at a time

stop once our val err begins getting worse

Called greedy because it makes choices that look best at the time.

Poll Everywhere

Think

1 min

pollev.com/cs416

Say you want to find the optimal two-feature model, using the forward stepwise algorithm. What model would the forward stepwise algorithm choose?

all subsets

forward stepwise

Features	Val Loss
# bath	201
# bed	300
sq ft	157
year built	224

Features	Val Loss
(# bath, # bed)	120
(# bath, sq ft)	131
(# bath, year built)	190
(# bed, sq ft)	137
(# bed, year built)	209
(sq ft, year built)	145

Poll Everywhere

Group

2 min

pollev.com/cs416

Say you want to find the optimal two-feature model, using the forward stepwise algorithm. What model would the forward stepwise algorithm choose?

Features	Val Loss
# bath	201
# bed	300
sq ft	157
year built	224

Features	Val Loss
(# bath, # bed)	120
(# bath, sq ft)	131
(# bath, year built)	190
(# bed, sq ft)	137
(# bed, year built)	209
(sq ft, year built)	145

Option 2
Regularization

Recap: Regularization

Before, we used the quality metric that minimize loss

$$\hat{w} = \underset{w}{\operatorname{argmin}}\, L(w)$$

Change quality metric to balance loss with measure of overfitting

- $L(w)$ is the measure of fit
- $R(w)$ measures the magnitude of coefficients

$$\hat{w} = \underset{w}{\operatorname{argmin}} \left[L(w) + R(w) \right]$$

How do we actually measure the magnitude of coefficients?

Recap: Magnitude

Come up with some number that summarizes the magnitude of the weights w.

$$\hat{w} = \underset{w}{\operatorname{argmin}}\, MSE(w) + \lambda R(w)$$

Sum?

$$R(w) = w_0 + w_1 + \cdots + w_d$$

Doesn't work because the weights can cancel out (e.g. $w_0 = 1000, w_1 = -1000$) which so $R(w)$ doesn't reflect the magnitudes of the weights

Sum of absolute values? ⇒ L1 Regularization

$$R(w) = |w_0| + |w_1| + \cdots + |w_d| = \|w\|_1$$

It works! We're using L1-norm, for L1-regularization (LASSO)

Sum of squares? ⇒ L2 Regularization

$$R(w) = |w_0|^2 + |w_1|^2 + \ \ldots + |w_d|^2 = w_0^2 + w_1^2 + \ \ldots + w_d^2 = \|w\|_2^2$$

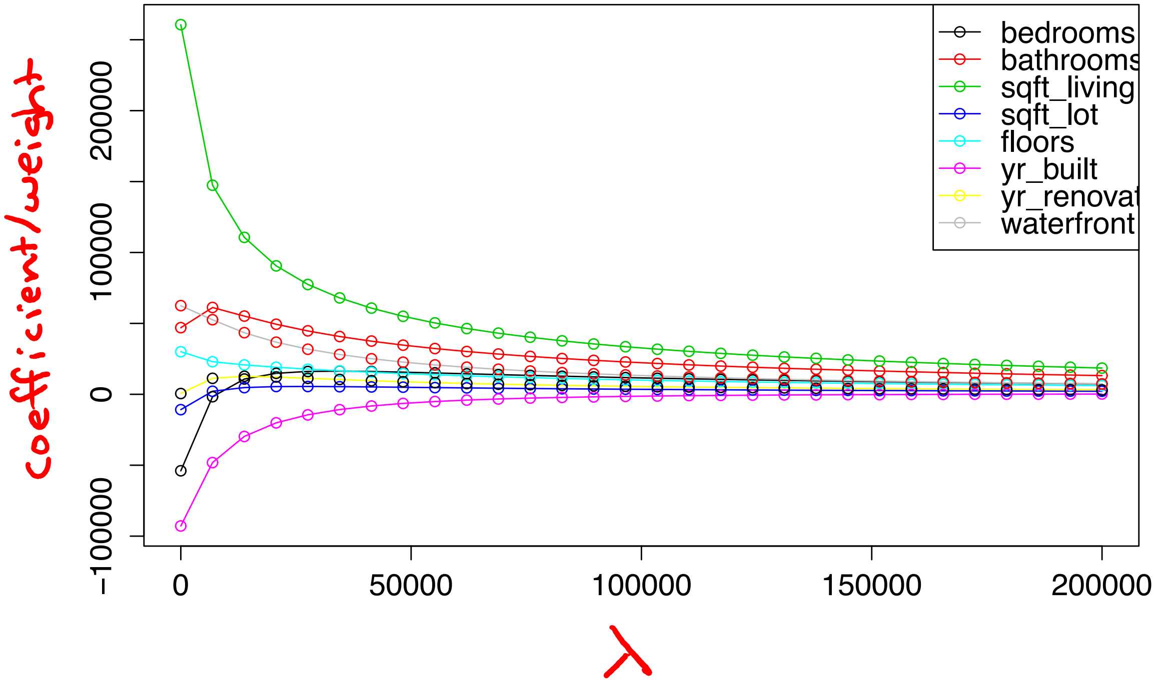
It works! We're using L2-norm, for L2-regularization (Ridge Regression)

Note: Definition of p-Norm: $\|w\|_p^p = |w_0|^p + |w_1|^p + \ \ldots + |w_d|^p$

Ridge for Feature Selection

MUST NORMALIZE

We saw that Ridge Regression shrinks coefficients, but they don’t become 0. What if we remove weights that are sufficiently small?

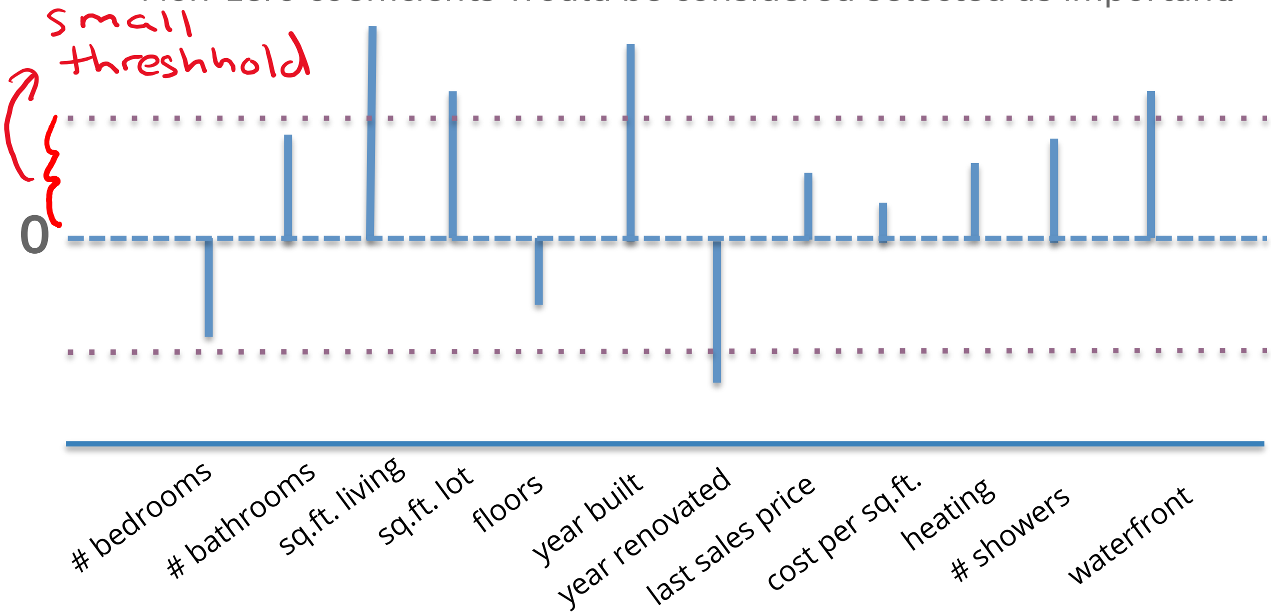


Ridge for Feature Selection

Instead of searching over a **discrete** set of solutions, use regularization to reduce coefficient of unhelpful features.

Start with a full model, and then "shrink" ridge coefficients near 0.

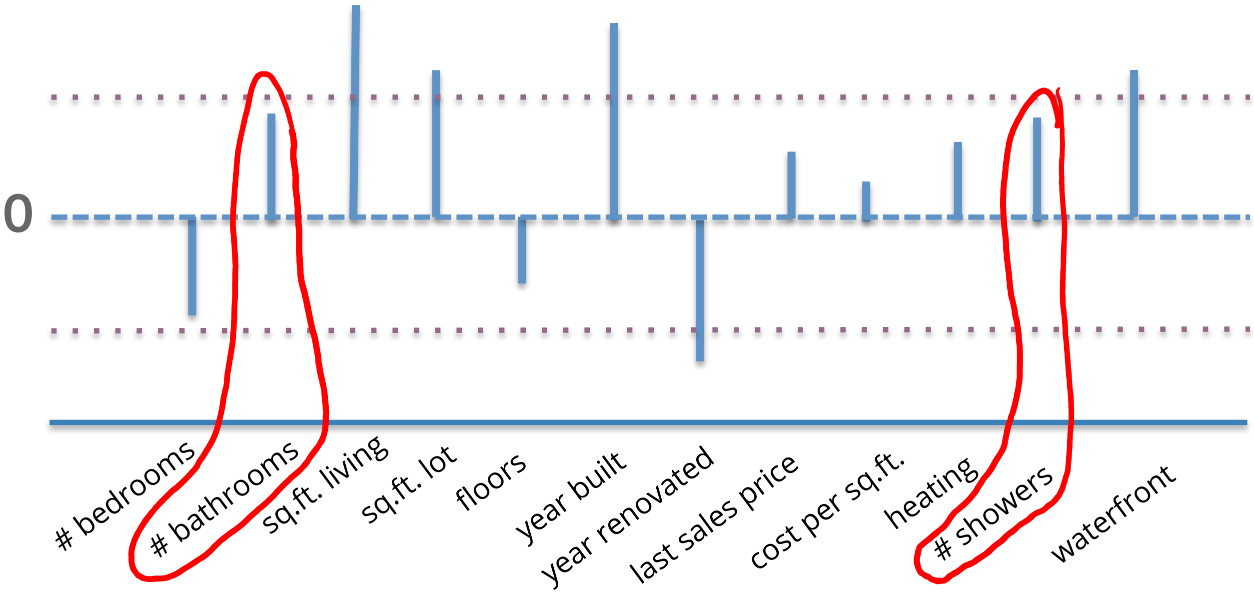
Non-zero coefficients would be considered selected as important.



Ridge for Feature Selection

Look at two related features #bathrooms and # showers.

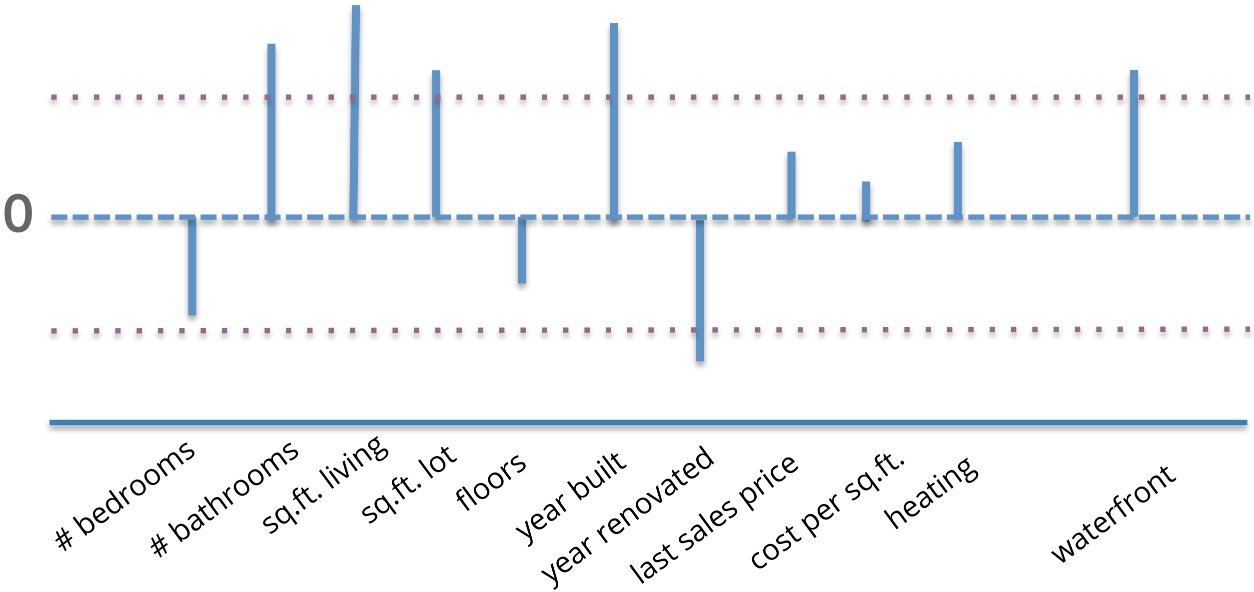
Our model ended up not choosing any features about bathrooms!



Ridge for Feature Selection

What if we had originally removed the # showers feature?

- The coefficient for # bathrooms would be larger since it wasn't "split up" amongst two correlated features
- Instead, it would be nice if there were a regularizer that favors sparse solutions in the first place to account for this…



LASSO Regression

Change quality metric to minimize

L1-norm

$\sum_{j=1}^{D} |w_j|$

$$\hat{w} = \underset{w}{\operatorname{argmin}}\, MSE(w) + \lambda||w||_1$$

λ is a tuning parameter that changes how much the model cares about the regularization term.

What if $\lambda = 0$?

$\hat{w} = \underset{w}{\operatorname{argmin}}\, MSE(w) = \hat{w}_{OLS}$

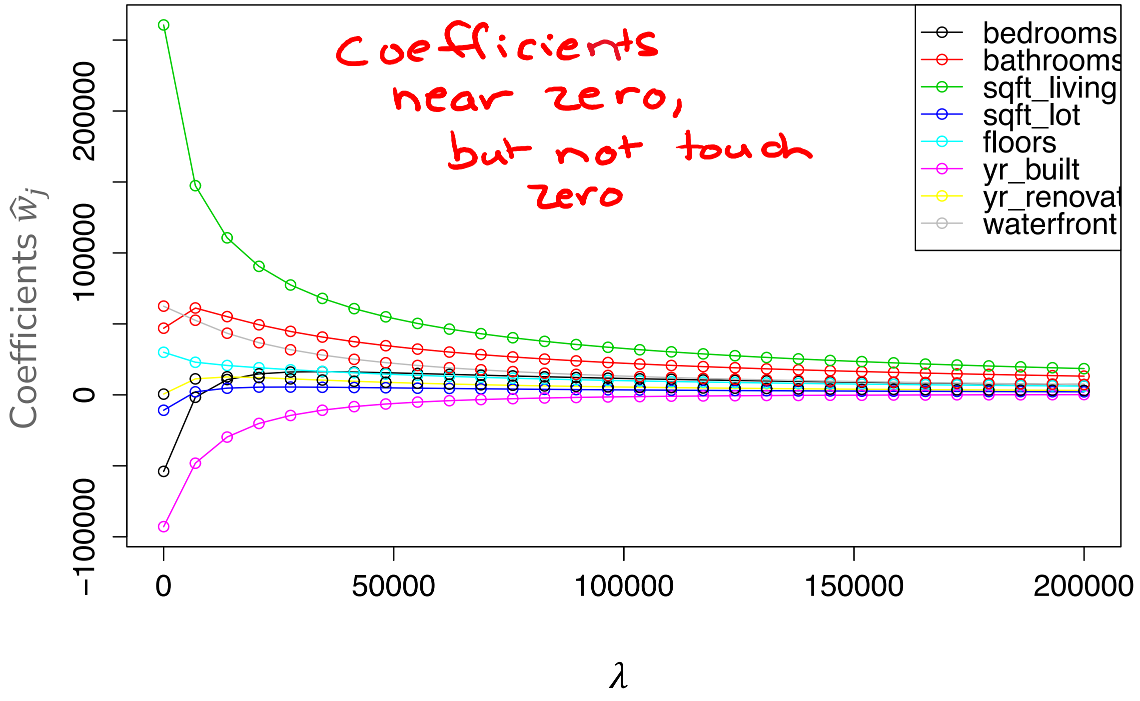
What if $\lambda = \infty$?

$\hat{w} = \underset{w}{\operatorname{argmin}}\, \lambda||w||_1 = \vec{0}$

λ in between?

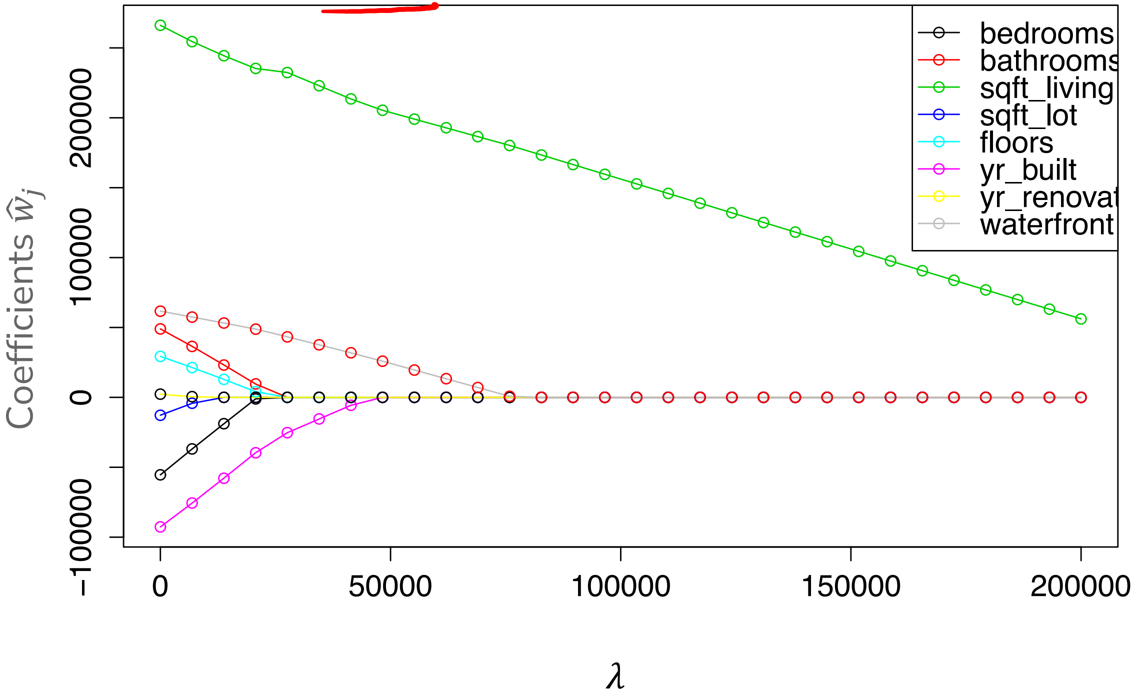
$0 \leq ||\hat{w}_{Lasso}||_1 \leq ||\hat{w}_{OLS}||_1$

Ridge (L2) Coefficient Paths



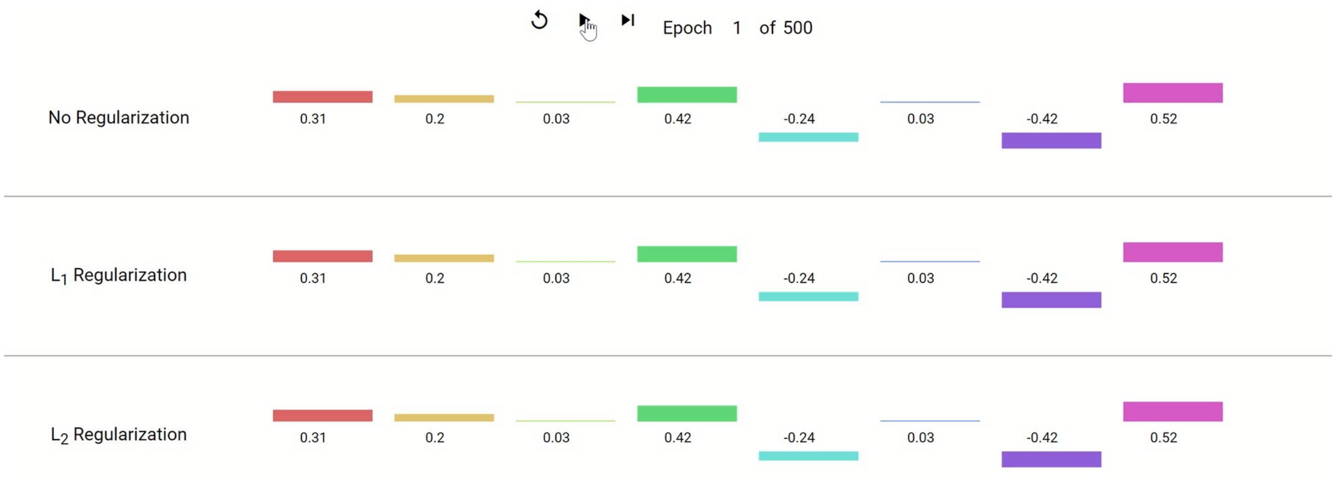
LASSO (L1) Coefficient Paths

less important features' coefficients will equal zero



Coefficient Paths – Another View

Example from Google's [Machine Learning Crash Course]



Demo

Similar demo to last time's with Ridge but using the LASSO penalty

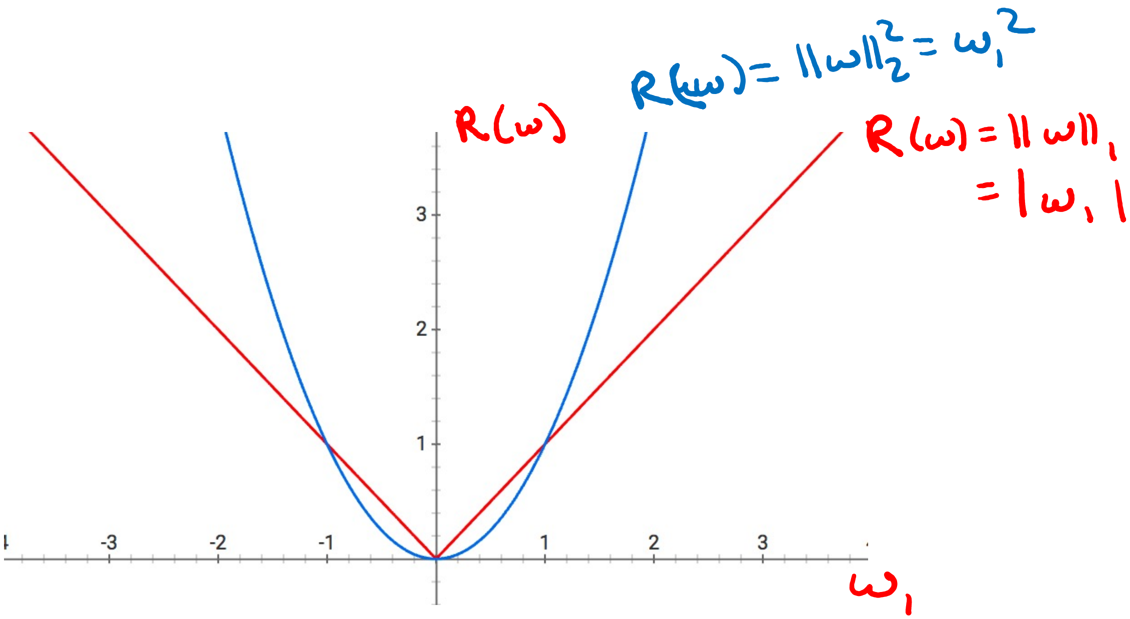
Poll Everywhere

Group

2 minutes

Why might the shape of the L1 penalty cause more sparsity than the L2 penalty?

$\omega = [\omega_0, \omega_1]$



Sparsity

$$\hat{w} = \underset{w}{\arg\min}\left[L(w) + \lambda R(w)\right]$$

When using the L1 Norm ($||w||_1$) as a regularizer, it favors solutions that are **sparse**. Sparsity for regression means many of the learned coefficients are 0.

$$w = [w_0, w_1]$$

This has to do with the shape of the norm

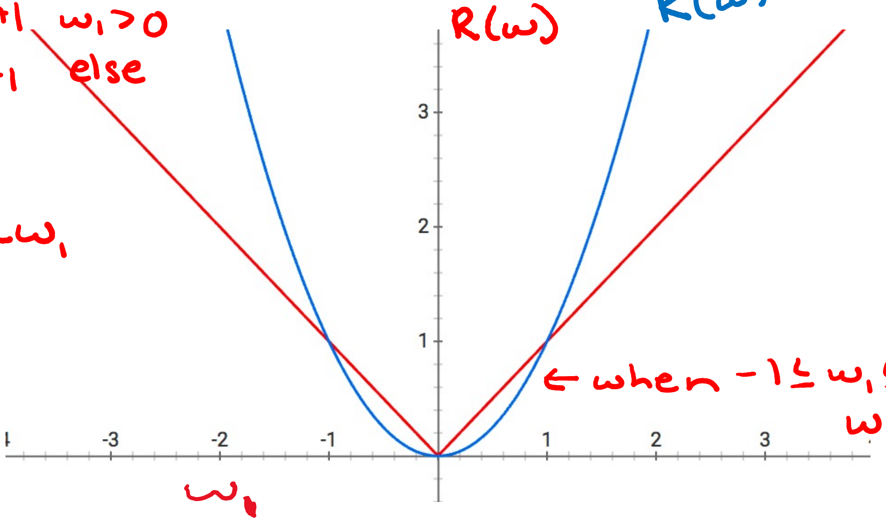
$$R(w) = \|w\|_2^2 = w_j^2$$

if $w_j = 10^{-2}$

$w_j^2 = 10^{-4}$

$$\frac{d}{dw_1}|w_1| = \begin{cases} +1 & w_1 > 0 \\ -1 & \text{else} \end{cases}$$

$$\frac{d}{dw_1} w_1^2 = 2w_1$$



← when $-1 \le w_1 \le 1$, then $w_1^2 \le |w_1|$

When w_j is small, w_j^2 is VERY small!

Sparsity Geometry

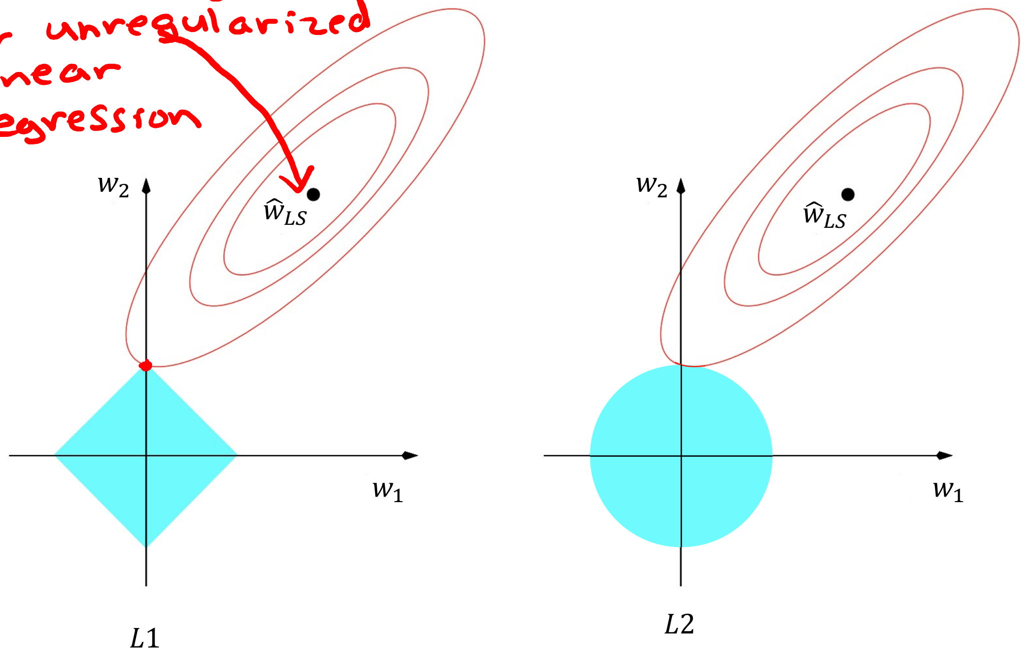
$\hat{w} = \underset{w}{\text{argmin}}\ MSE(w)$

such that

$R(w) \leq \beta$

Another way to visualize why LASSO prefers sparse solutions

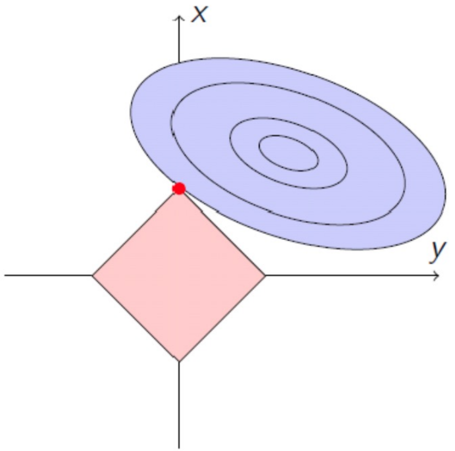
optimal weigths for unregularized Linear Regression



The L1 ball has spikes (places where some coefficients are 0)

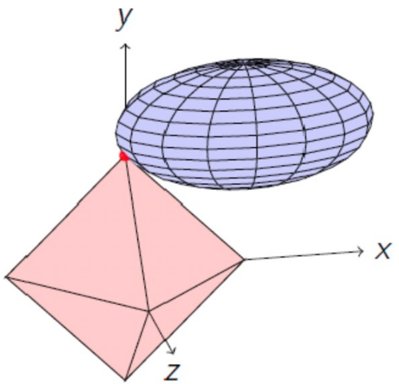
Sparsity Geometry

2D



L1

3D



L1

Brain Break

Poll Everywhere

Think

1 min

pollev.com/cs416

How should we choose the best value of λ for LASSO?

a) Pick the λ that has the smallest $MSE(\hat{w})$ on the **validation set**

b) Pick the λ that has the smallest $MSE(\hat{w}) + \lambda||\hat{w}||_2^2$ on the **validation set**

c) Pick the λ that results in the most zero coefficients

d) Pick the λ that results in the fewest zero coefficients

e) None of the above

Same as in Ridge!

Choosing λ

Exactly the same as Ridge Regression :)

This will be true for almost every **hyper-parameter** we talk about

A **hyper-parameter** is a parameter you specify for the model that influences which parameters (e.g. coefficients) are learned by the ML aglorithm

For any hyper-parameter:

Pick the hyperparameter tha has the lowest $err_{val} = MSE_{val}(\hat{w})$

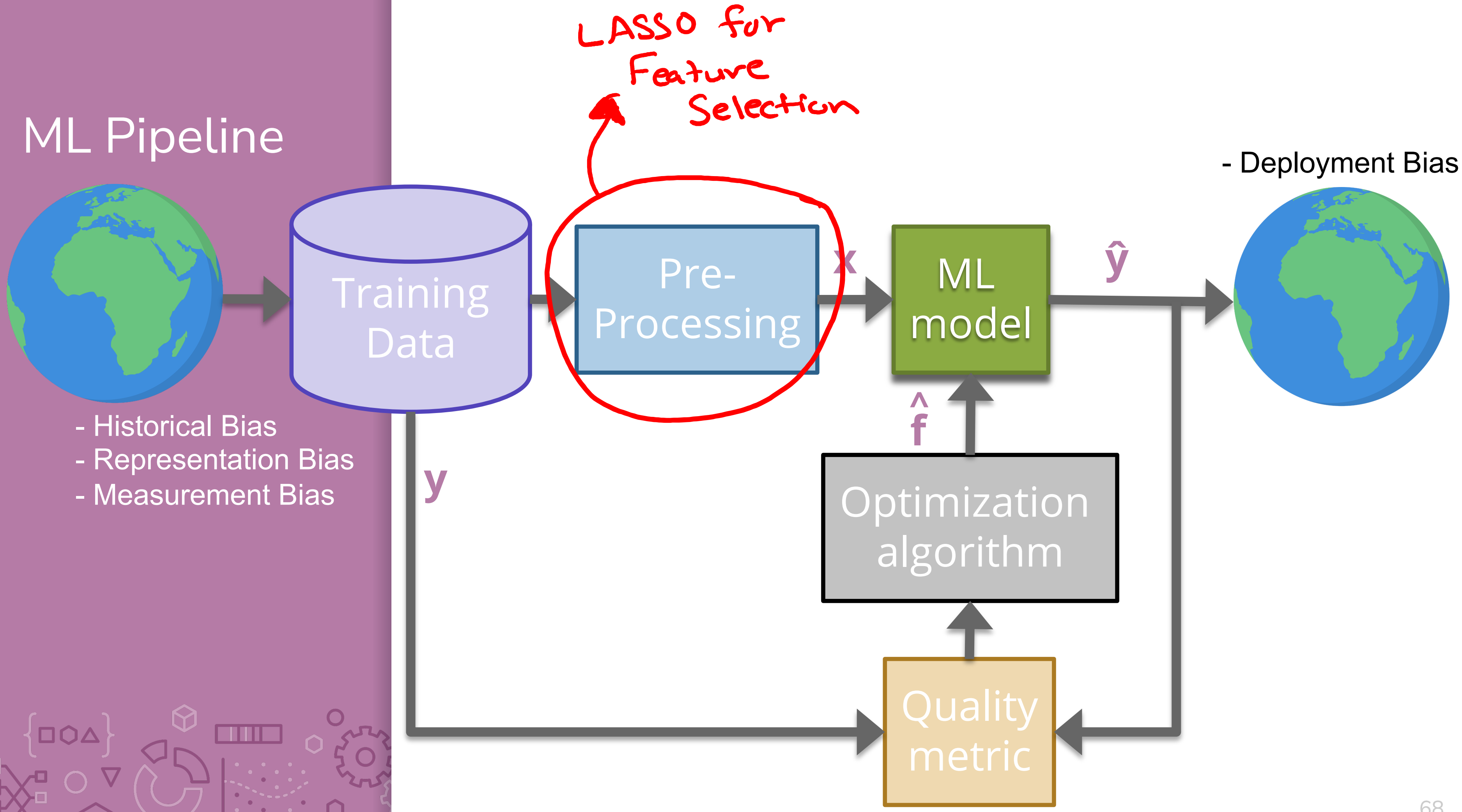
LASSO in Practice

A very common usage of LASSO is in feature selection. If you have a model with potentially many features you want to explore, you can use LASSO on a model with all the features and choose the appropriate λ to get the right complexity.

Then once you find the non-zero coefficients, you can identify which features are the most important to the task at hand*

* e.g., using domain-specific expertise

ML Pipeline


LASSO for Feature Selection
Training Data
Pre-Processing
x
ML model
ŷ
- Deployment Bias
- Historical Bias
- Representation Bias
- Measurement Bias
y
f̂
Optimization algorithm
Quality metric

As you increase λ, the resulting models have less variance and high bias

De-biasing LASSO

LASSO (and Ridge) adds bias to the Least Squares solution (this was intended to avoid the variance that leads to overfitting)

Recall Bias-Variance Tradeoff

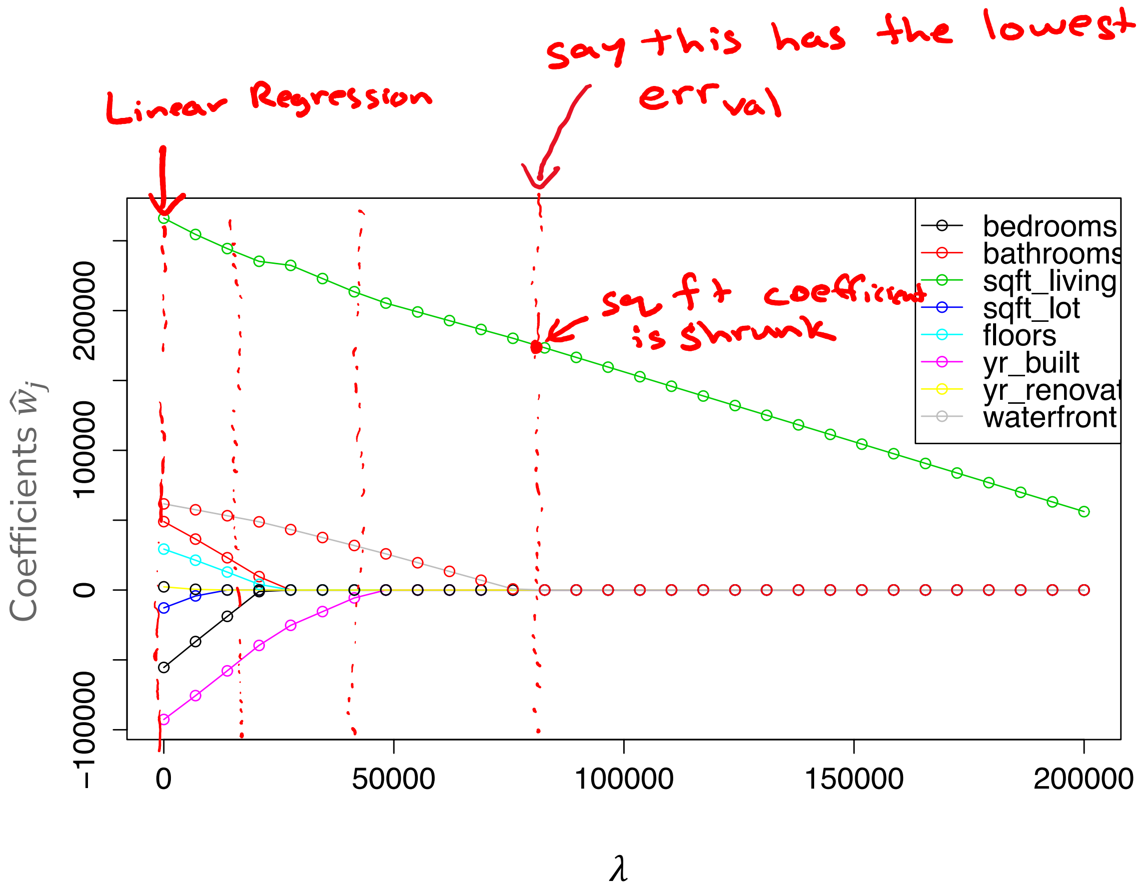
It's possible to try to remove the bias from the LASSO solution using the following steps

1. Run LASSO to select which features should be used (those with non-zero coefficients)
2. Run regular Ordinary Least Squares on the dataset with only those features

→ Linear Regression

Coefficients are no longer shrunk from their true values

LASSO (L1) Coefficient Paths



(De-biased) LASSO In Practice

compute μ, σ

pre-processing

1. Split the dataset into train, (val), and test sets
2. Normalize features. Fit the normalization on the train set. apply that normalization on the train, (val), and test sets.
3. Use validation or cross-validation to find the value of λ that that results in a LASSO model with the lowest validation error.
4. Select the features of that model that have non-zero weights.
5. Train a Linear Regression model with those features.
6. Evaluate on the test set.

Issues with LASSO

1. Within a group of highly correlated features (e.g. # bathroom and # showers), LASSO tends to select amongst them arbitrarily.
   - Maybe it would be better to select them all together?
2. Often, empirically Ridge tends to have better predictive performance

Elastic Net aims to address these issues

$$\hat{w}_{ElasticNet} = \underset{w}{\operatorname{argmin}} \, MSE(w) + \lambda_1 \left|\left|w\right|\right|_1 + \lambda_2 \left|\left|w\right|\right|_2^2$$

Combines both to achieve best of both worlds!

Poll Everywhere

Think

1 min

pollev.com/cs416

Suppose you wanted to try out the following models:

LASSO with hyperparameter choices $\lambda \in [0.01, 1, 10]$

Ridge with hyperparameter choices $\lambda \in [0.05, 5, 50]$

Of the 6 models you will try, how do you pick the best predictor learned?

a) Pick the predictor that has the smallest $MSE(\hat{w})$ on the **validation set**

b) Pick the predictor that has the smallest $MSE(\hat{w}) + \underline{\lambda||\hat{w}||_2^2}$ on the **validation set**

c) Pick the predictor that has the smallest $MSE(\hat{w}) + \underline{\lambda||\hat{w}||_1}$ on the **validation set**

d) None of the above

ONLY USED IN TRAINING!

A Big Grain of Salt

Be careful when interpreting the results of feature selection or feature importance in Machine Learning!

- Selection only considers features included
- Sensitive to correlations between features
- Results depend on the algorithm used!

At the end of the day, the best models combine statistical insights with domain-specific expertise!

Differences between L1 and L2 regularizations

L1 (LASSO):

- Introduces more sparsity to the model
- Less sensitive to outliers
- Helpful for feature selection, making the model more interpretable
- More computational efficient as a model (due to the sparse solutions, so you have to compute less dot products)

L2 (Ridge):

- Makes the weights small (but not 0)
- More sensitive to outliers (due to the squared terms)
- Usually works better in practice

Recap

Theme: Using regularization to do feature selection

Ideas:

Describe "all subsets" approach to feature selection and why it's impractical to implement.

Formulate LASSO objective

Describe how LASSO coefficients change as hyper-parameter λ is varied

Interpret LASSO coefficient path plot

Compare and contrast LASSO (L1) and Ridge (L2)

Poll Everywhere

Think

5 mins pair
5 mins share

What is a problem relevant to your field / interests that you'd want to use (or are using) machine learning to solve?

Discuss in groups of ~3.