① # of features/basis functions

② $+ \lambda \|w\|_2^2$

feature/

Simple variable selection: LASSO for sparse regression

Sparsity

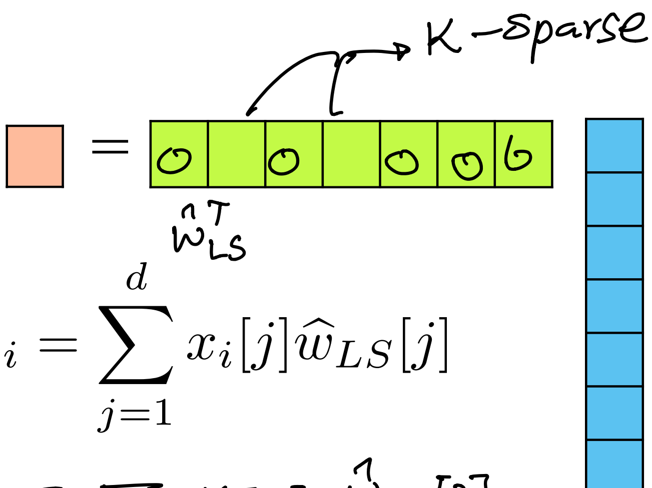
$$\widehat{w}_{LS} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2$$

- Vector w is **sparse**, if many entries are zero

Sparsity

$$\widehat{w}_{LS} = \arg\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2$$

- Vector w is **sparse**, if many entries are zero
  - **Efficiency**: If size(w) = 100 Billion, each prediction $w^T x$ is expensive:
    - If w is sparse, prediction computation only depends on number of non-zeros in w



$$\widehat{y}_i = \widehat{w}_{LS}^\top x_i = \sum_{j=1}^d x_i[j]\widehat{w}_{LS}[j]$$

$$= \sum_{j:\ \hat{w}_{LS} \text{ is non-zero}} x_i[j] \cdot \hat{w}_{LS}[j] \quad : O(K) \text{ computations}$$

Sparsity

$$\widehat{w}_{LS} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2$$

- Vector w is **sparse**, if many entries are zero
  - **Interpretability**: What are the relevant features to make a prediction?

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

- How do we find "best" subset of features useful in predicting the price among all possible combinations?

Finding best subset: Exhaustive

> Try all subsets of size $\overbrace{1, 2, 3, \ldots}^{d}$ and one that minimizes validation error

Minimum Description length.

> Problem?

$$\sum_{k=1}^{d} \binom{d}{k} = 2^d$$

$$\text{Error}_{cv} + \underline{\underline{\lambda}} \cdot \underbrace{\|w\|_0}_{\#\text{ non-zero in } w,}$$

Finding best subset: Greedy

Forward stepwise:
Starting from simple model and iteratively add features most useful to fit

Backward stepwise:
Start with full model and iteratively remove features least useful to fit

Combining forward and backward steps:
In forward algorithm, insert steps to remove features no longer as important

Lots of other variants, too.

Forward Greedy

$T = \emptyset$

for $j = 1, \ldots, K$

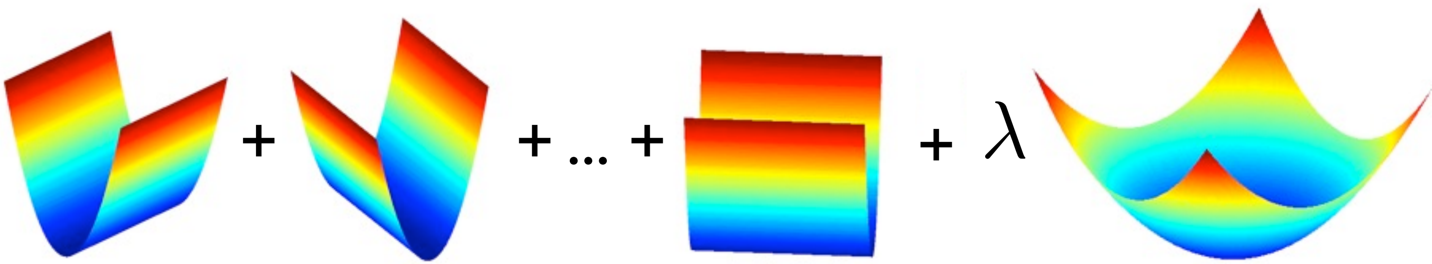
$i^* \leftarrow \arg\min_{w} \text{Error}_{cv}(\text{using } T \cup \{i\})$

$T = T \cup \{i^*\}$

Finding best subset: Regularize

Ridge regression makes coefficients small

$$\widehat{w}_{ridge} = \arg\min_w \underbrace{\sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2}_{\text{LS-error.}} + \lambda ||w||_2^2$$

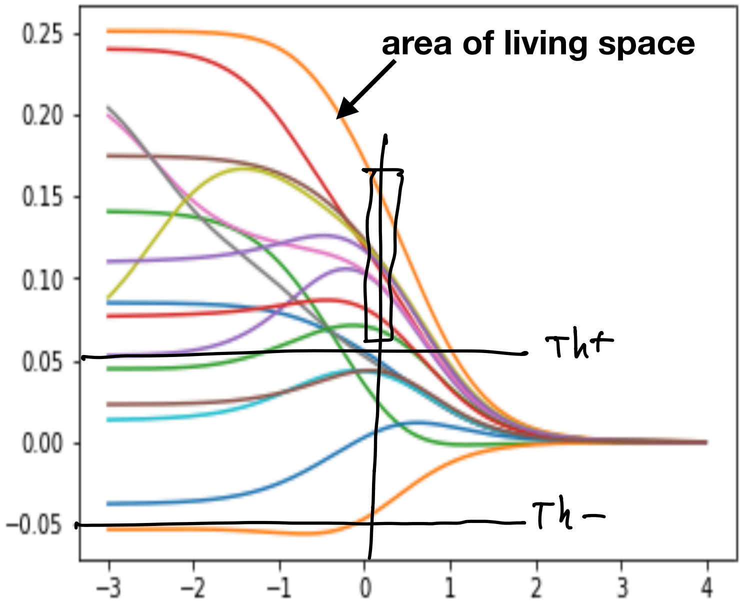


Finding best subset: Regularize

Ridge regression makes coefficients small

$$\hat{w}_{ridge} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

w_i's

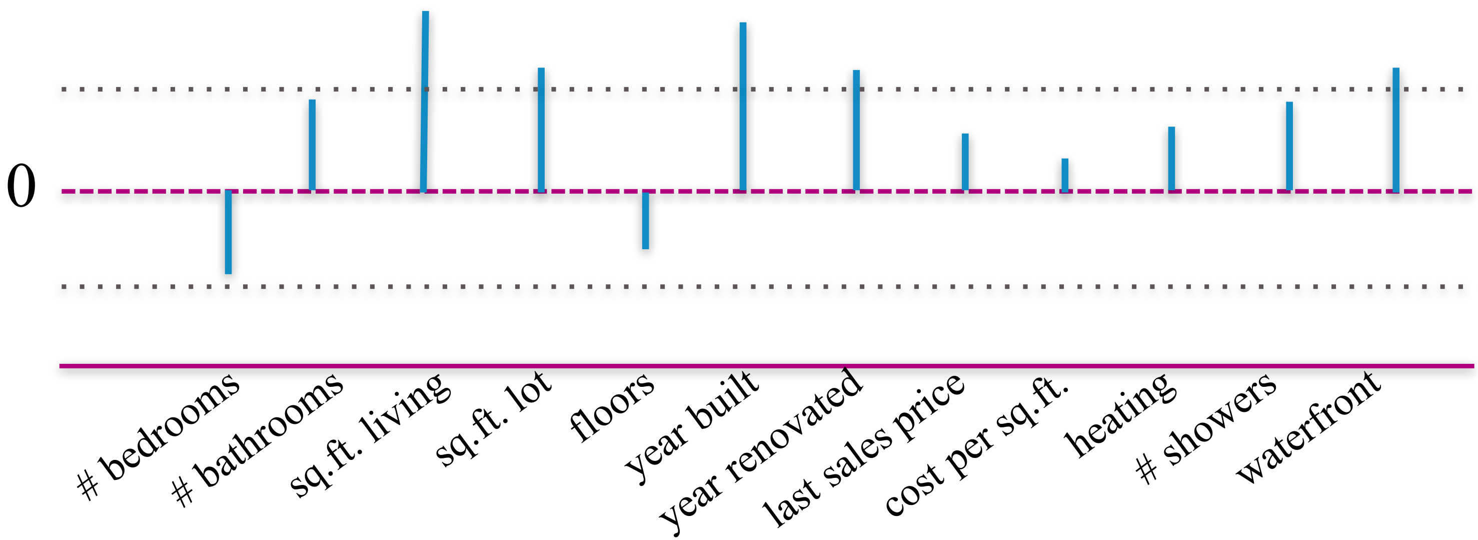


$\log_{10}(\lambda)$

Thresholded Ridge Regression

$$\widehat{w}_{ridge} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

Why don't we just set **small** ridge coefficients to 0?



Thresholded Ridge Regression

suppose that all houses in training had same # of bathrooms as # of showers.

prediction is same

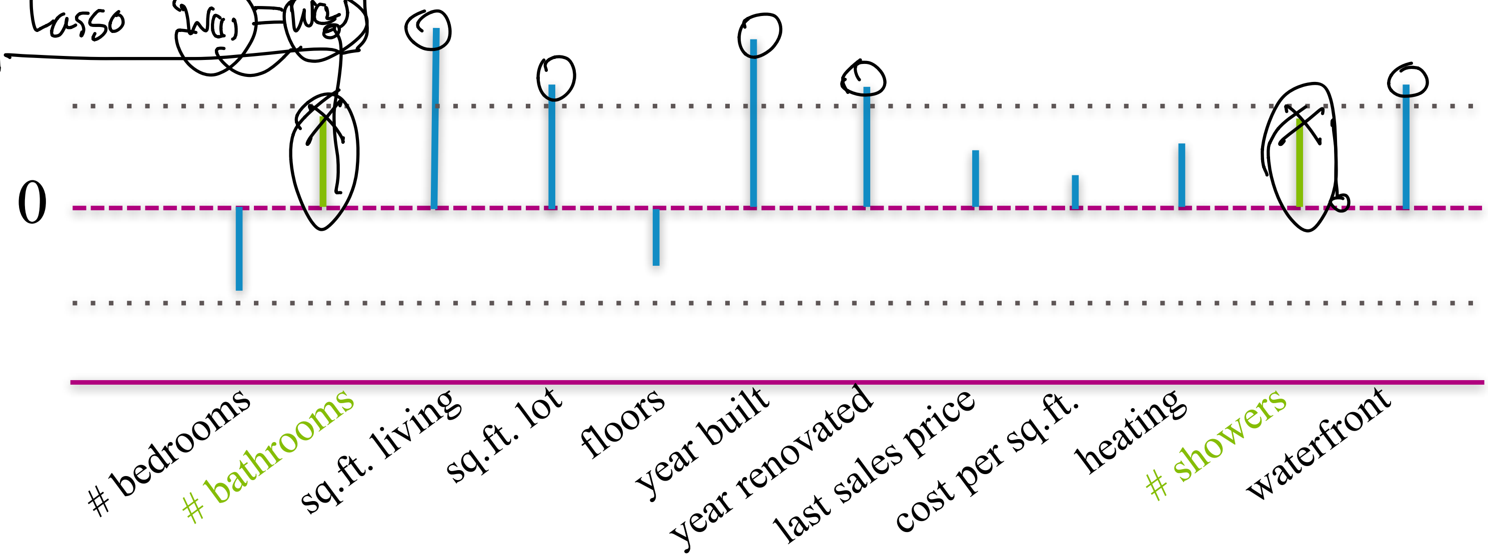
$w(1)$: $w_{bath} = 1$, $w_{shower} = 1$ → regularization $1^2 + 1^2 = 2$ → $\boxed{1+1=2}$

$w(2)$: $w_{bath} = 2$, $w_{shower} = 0$ → $2^2 = 4$ → $\boxed{2+0=2}$

$$w_{ridge} = \arg\min_w \sum_{i=1}^{n} (y_i - x_i^T w)^2 + \lambda ||w||_2^2$$

Ridge	$w(1) > w(2)$,
Lasso	$w(1) = w(2)$

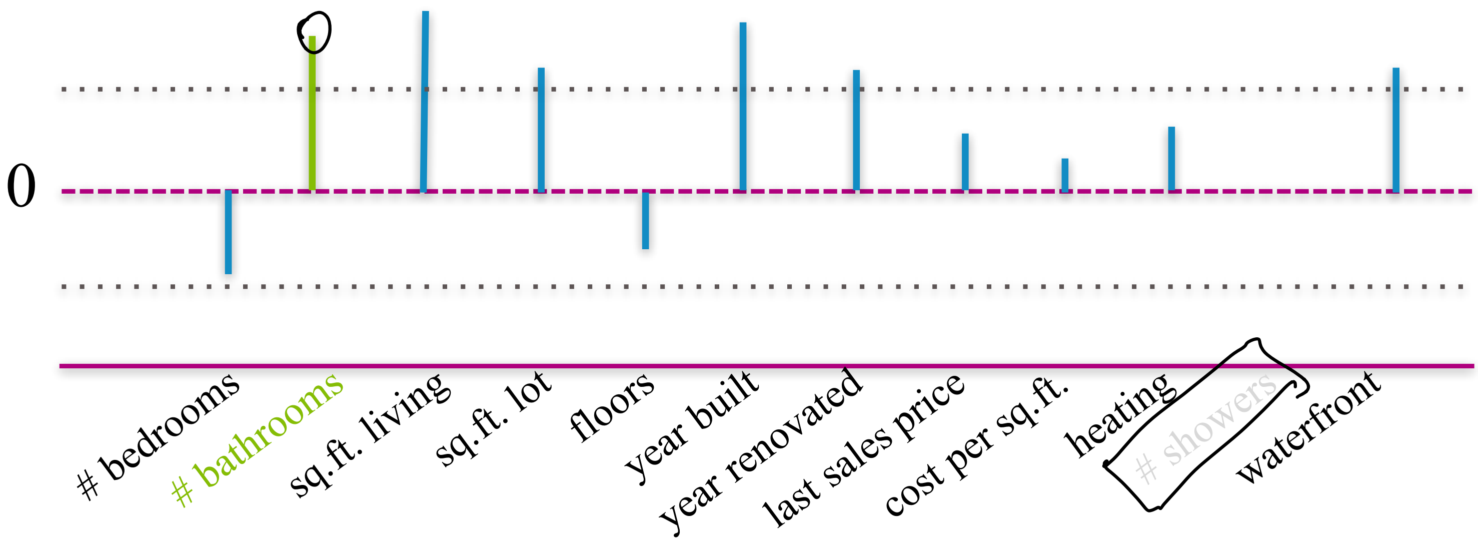
Consider two related features (bathrooms, showers)



Thresholded Ridge Regression

$$\widehat{w}_{ridge} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

What if we **didn't** include showers? Weight on bathrooms increases!

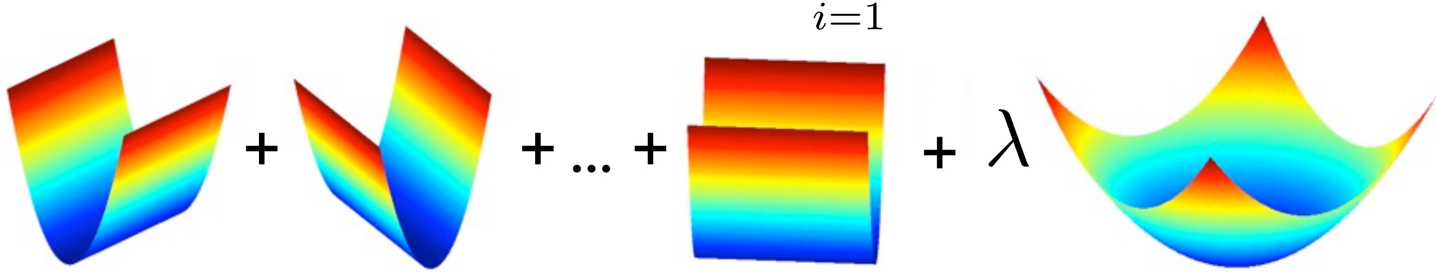


Can another regularizer perform selection automatically?

Recall Ridge Regression

- Ridge Regression objective:

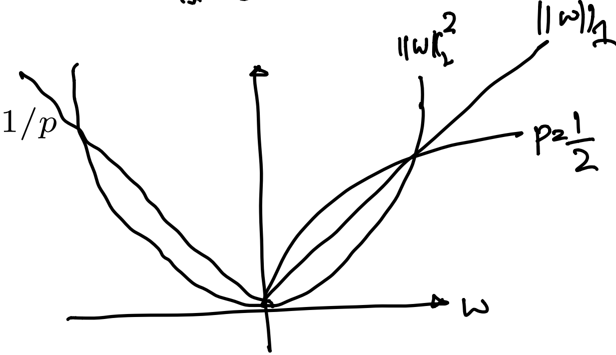
$$\widehat{w}_{ridge} = \arg\min_{w} \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

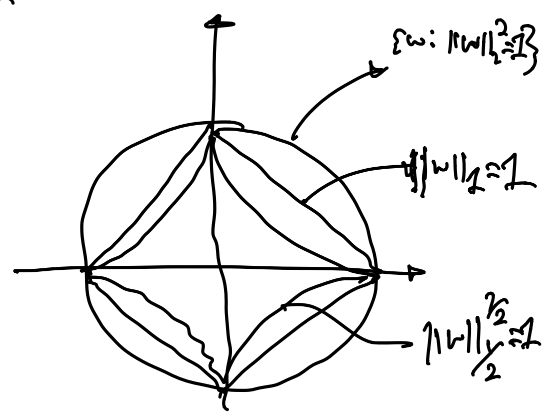


1-D

2-D

$$||w||_p = \left(\sum_{i=1}^{d} |w_i|^p\right)^{1/p}$$





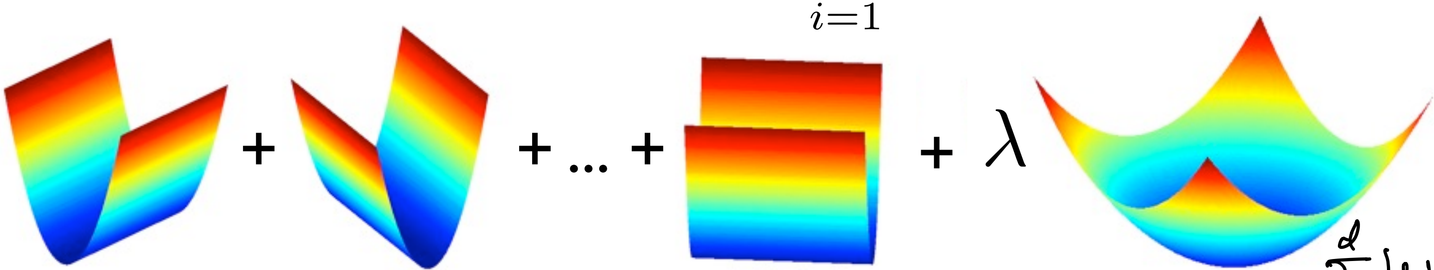
$||w||_1 = |w_1| + |w_2|$

Ridge vs. Lasso Regression

$\|w\|_p = \left(\sum_{j=1}^{d} (|w_j|^p) \right)^{1/p}$

- Ridge Regression objective:

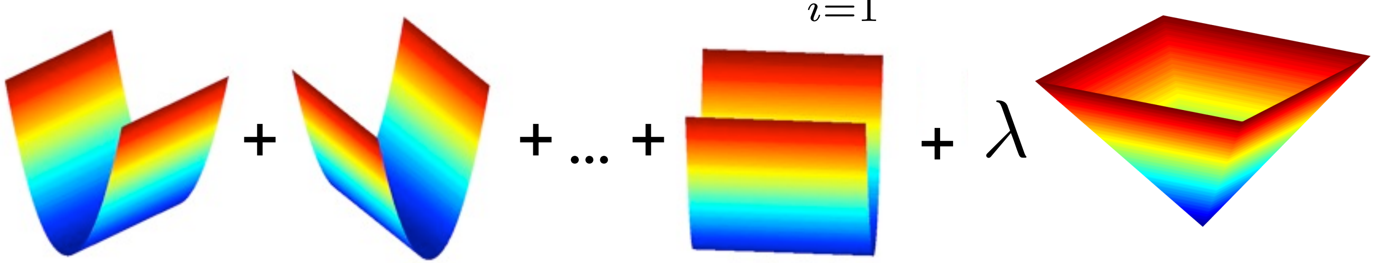
$$\widehat{w}_{ridge} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$



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

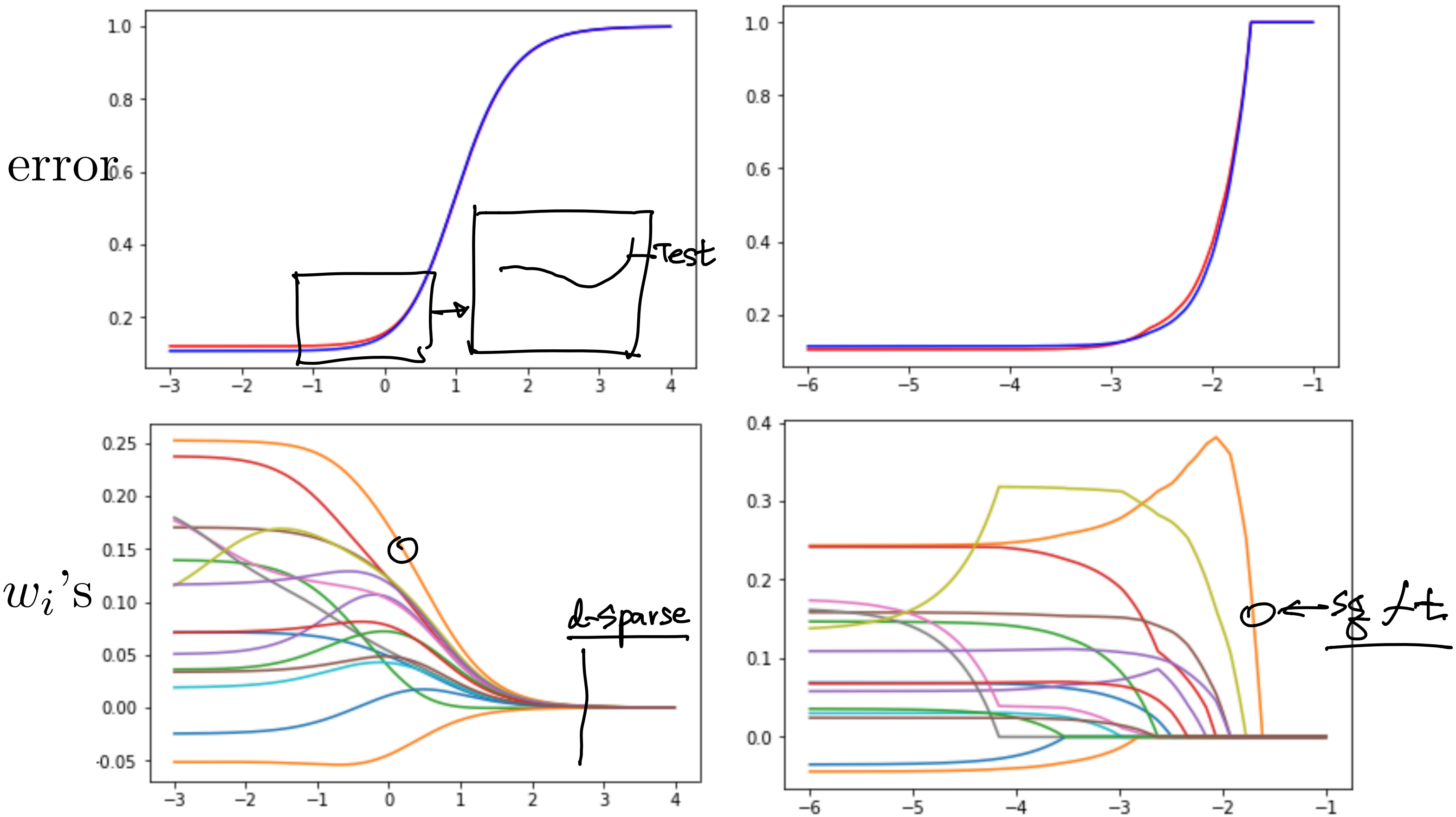
- Lasso objective:

$$\widehat{w}_{lasso} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \boxed{\lambda ||w||_1}$$



Example: house price with 16 features

test error is red and train error is blue



$\log_{10} \lambda$ Ridge regression

$\log_{10} \lambda$ Lasso regression

Lasso regression naturally gives sparse features

- **feature selection** with Lasso regression
  1. choose λ based on cross validation error
  2. keep only those features with non-zero (or not-too-small) parameters in w at optimal λ $\longrightarrow$ feature selection
  3. **retrain** with the sparse model and $\underline{\lambda = 0}$

     No Regularization.

Example: piecewise-linear fit

- We use Lasso on the piece-wise linear example

$$
\begin{aligned}
h_0(x) &= 1 \\
h_i(x) &= [x + 1.1 - 0.1i]^+
\end{aligned}
$$

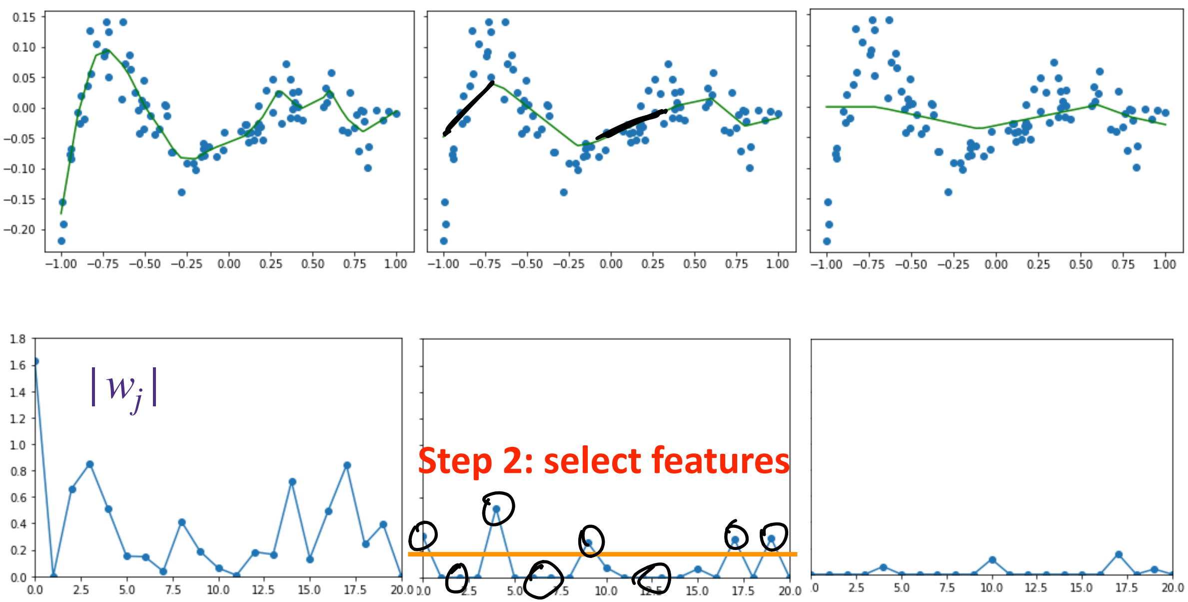
Step 1: find optimal λ^*

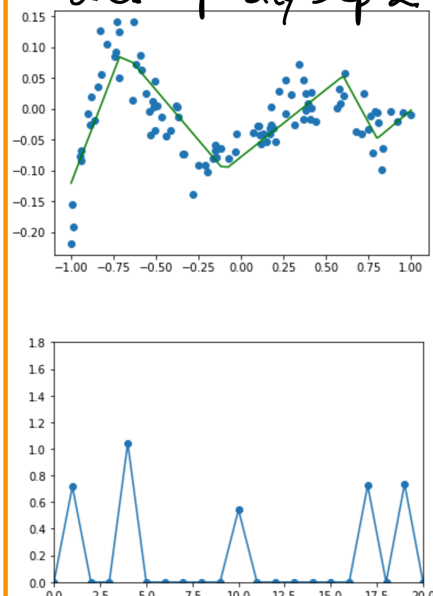
$$\text{minimize}_w \ \ \mathscr{L}(w) + \lambda \|w\|_1$$

Step 3: retrain

$$\text{minimize}_w \ \ \mathscr{L}(w)$$

s.t. sparsity step-2





$\lambda = 10^{-8}$ $\lambda = 10^{-4}$ $\lambda = 2 \times 10^{-4}$

$\lambda = 0$

but only use selected features

- de-biasing (via re-training) is critical!

Penalized Least Squares

$$\text{Ridge} : r(w) = ||w||_2^2 \qquad \text{Lasso} : r(w) = ||w||_1$$

$$\widehat{w}_r = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda r(w)$$

Penalized Least Squares

$$\text{Ridge} : r(w) = ||w||_2^2 \qquad \text{Lasso} : r(w) = ||w||_1$$

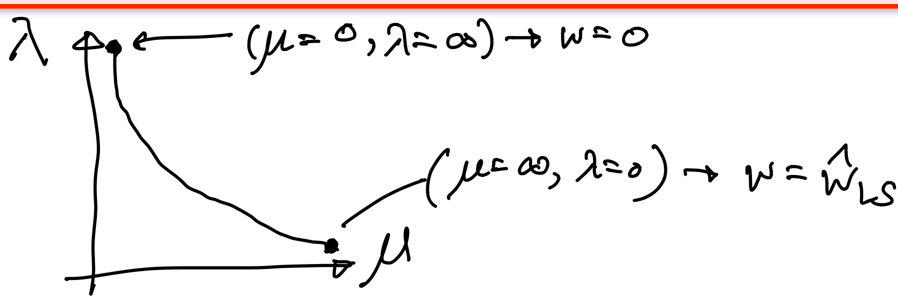
$$\widehat{w}_r = \arg\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2 + \lambda r(w)$$ ← Penalized

$\hat{w}_\lambda = \hat{w}_\mu$

For any $\lambda \geq 0$ for which $\hat{w}_r$ achieves the minimum, there exists a $\mu \geq 0$ such that

$$\widehat{w}_r = \arg\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2 \qquad \text{subject to } r(w) \leq \mu$$

Constrained

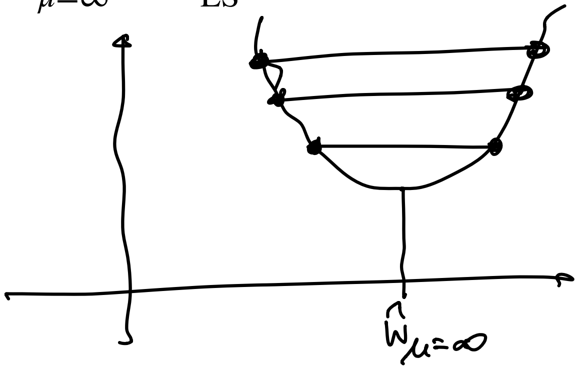


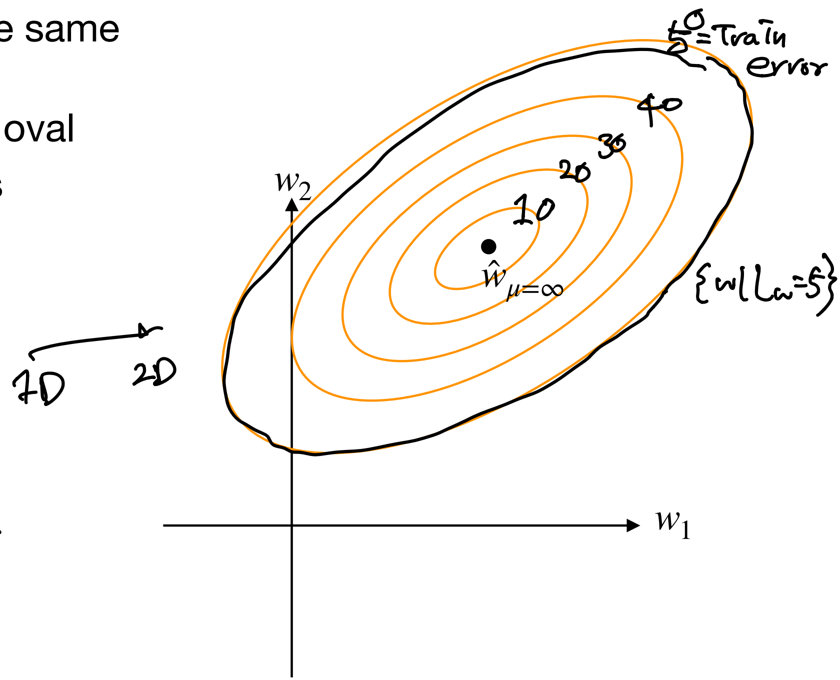
Why does Lasso give sparse solutions?

$$\text{minimize}_w \quad \sum_{i=1}^{n} (w^T x_i - y_i)^2$$

$$\text{subject to} \quad \|w\|_1 \leq \mu$$

- the **level set** of a function $\mathscr{L}(w_1, w_2)$ is defined as the set of points (w_1, w_2) that have the same function value
- the level set of a quadratic function is an oval
- the center of the oval is the least squares solution $\hat{w}_{\mu=\infty} = \hat{w}_{\text{LS}}$



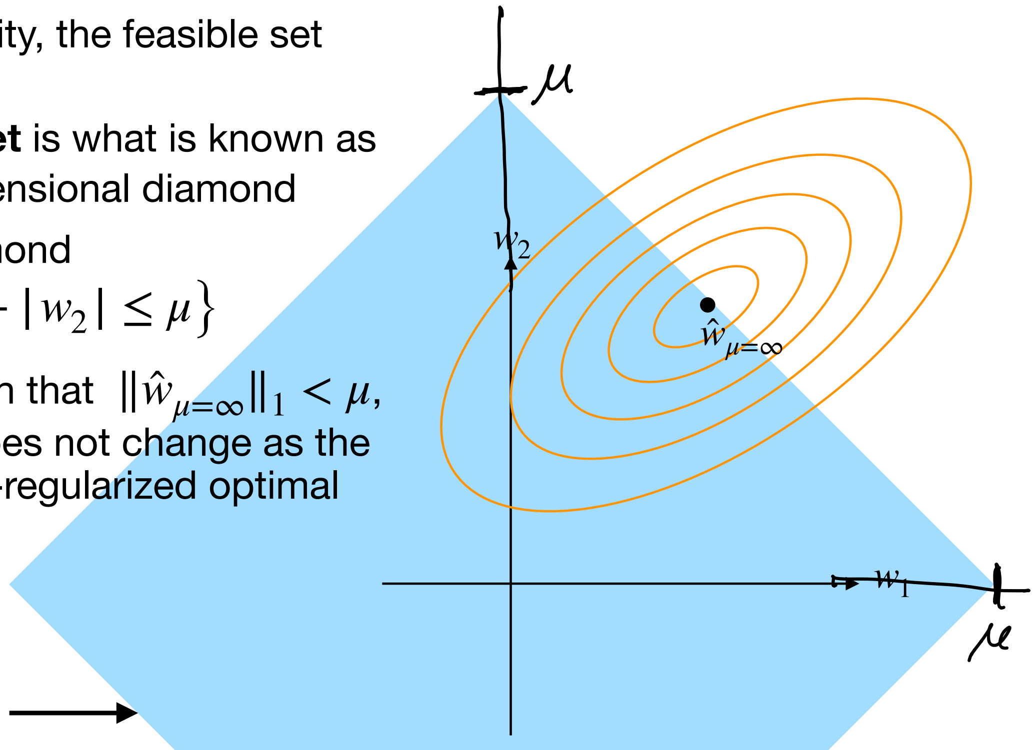


Why does Lasso give sparse solutions?

$$\text{minimize}_w \quad \sum_{i=1}^{n} (w^T x_i - y_i)^2$$

$$\text{subject to} \quad \|w\|_1 \;\leq\; \mu$$

- as we decrease μ from infinity, the feasible set becomes smaller
- the shape of the **feasible set** is what is known as L_1 ball, which is a high dimensional diamond
- In 2-dimensions, it is a diamond
$$\left\{ (w_1, w_2) \,\middle|\, |w_1| + |w_2| \leq \mu \right\}$$
- when μ is large enough such that $\|\hat{w}_{\mu=\infty}\|_1 < \mu$, then the optimal solution does not change as the feasible set includes the un-regularized optimal solution



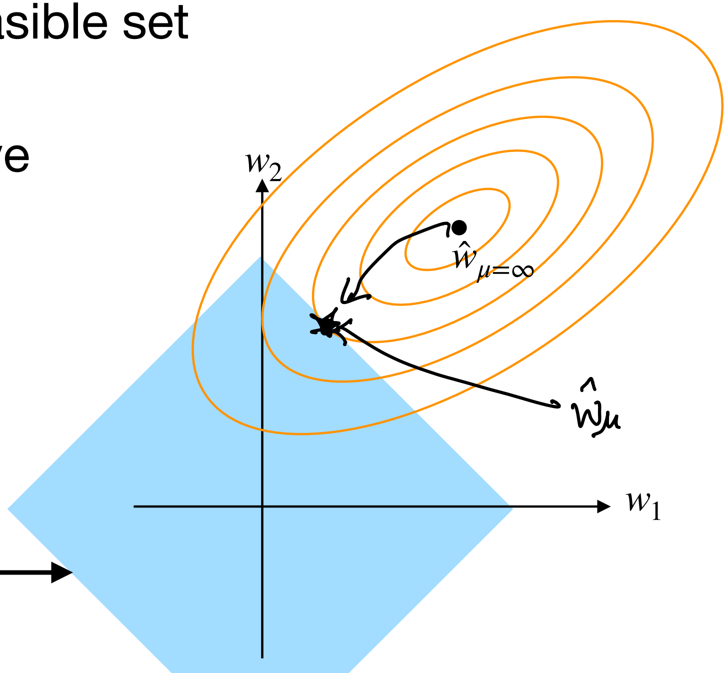
feasible set: $\{w \in \mathbb{R}^2 \mid \|w\|_1 \leq \mu\}$

Why does Lasso give sparse solutions?

$$\text{minimize}_w \quad \sum_{i=1}^{n} (w^T x_i - y_i)^2$$

$$\text{subject to} \quad \|w\|_1 \leq \mu$$

- As μ decreases (which is equivalent to increasing regularization) the feasible set (blue diamond) shrinks
- The optimal solution of the above optimization is



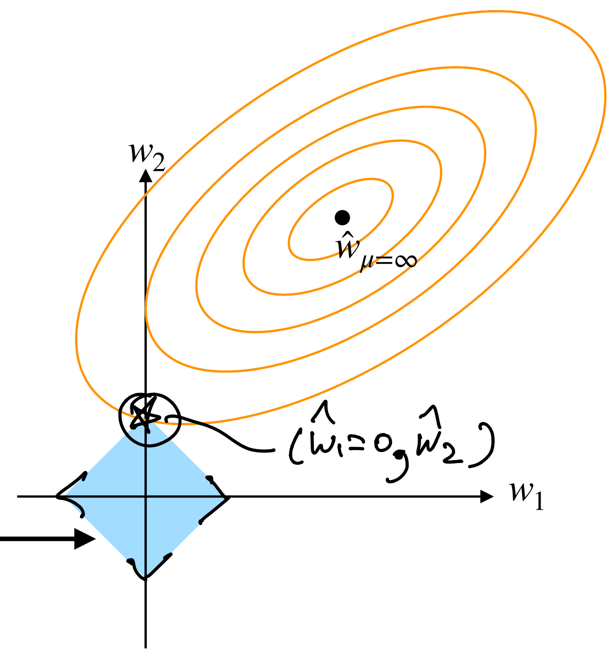
feasible set: $\{w \in \mathbb{R}^2 \mid \|w\|_1 \leq \mu\}$

Why does Lasso give sparse solutions?

$$\text{minimize}_w \quad \sum_{i=1}^{n} (w^T x_i - y_i)^2$$

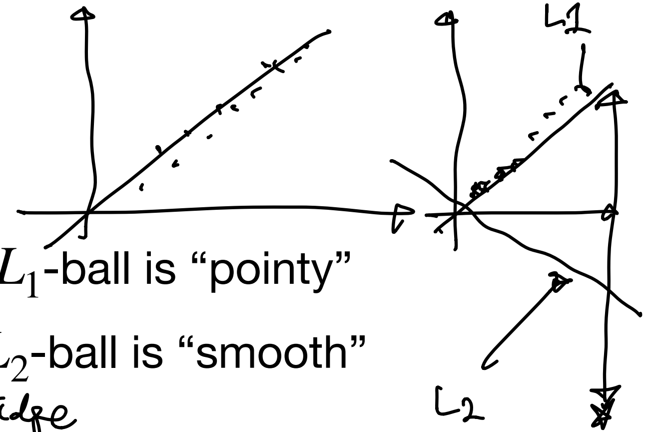
$$\text{subject to} \quad \|w\|_1 \leq \mu$$

- For small enough μ, the optimal solution becomes **sparse**
- This is because the L_1-ball is "pointy",i.e., has sharp edges aligned with the axes

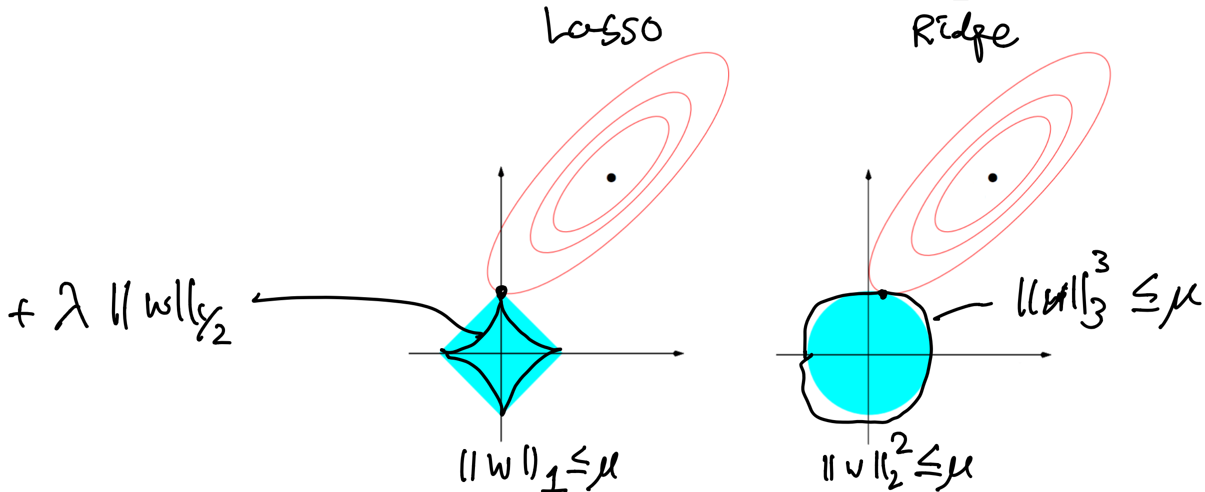


feasible set: $\{w \in \mathbb{R}^2 \mid \|w\|_1 \leq \mu\}$

Penalized Least Squares



- Lasso regression finds sparse solutions, as L_1-ball is "pointy"
- Ridge regression finds dense solutions, as L_2-ball is "smooth"



$$\text{minimize}_w \quad \sum_{i=1}^{n} (w^T x_i - y_i)^2 \qquad\qquad \text{minimize}_w \quad \sum_{i=1}^{n} (w^T x_i - y_i)^2$$

$$\text{subject to} \quad \|w\|_1 \leq \mu \qquad\qquad \text{subject to} \quad \|w\|_2^2 \leq \mu$$

Questions?

vs.

$\min \; L_w(x, y)$ $\quad$ $\min \; L_w(x, y)$

s.t. $\|w\|_1 \leq 1$ $\quad$ s.t. $\|w\|_{1/2} \leq 1$

less sparse. $\quad$ more sparse

$\hat{w}_1$ $\quad$ $\hat{w}_{1/2}$

more zeros.

k-sparse $=$ k non-zero

more sparse $= k \downarrow =$ more zero.