CSE 446
Sparsity & the LASSO

Natasha Jaques

UNIVERSITY of WASHINGTON

Motivation

Can we make our model more compact and interpretable?

Sparsity

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

- We learned to measure ***sensitivity*** by the size of weights: $\|w\|_2^2$
- Vector w is ***sparse***, if many entries are zero
  - A vector w is said to be k-sparse if at most k entries are non-zero
  - We are interested in k-sparse w with $k \ll d$
  - Why do we prefer sparse vector w in practice?

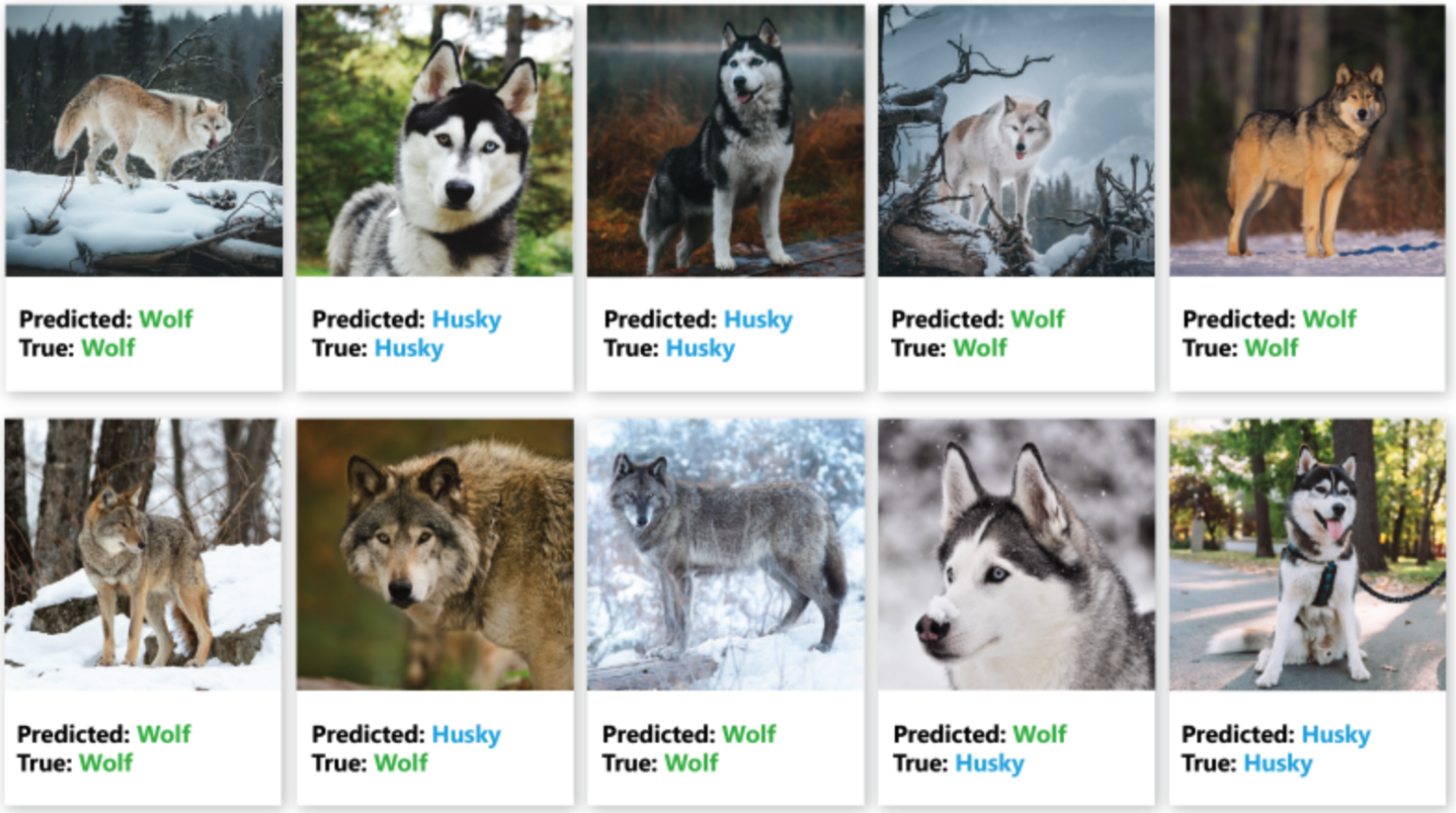
Computational efficiency

Interpretability

Remove spurious features

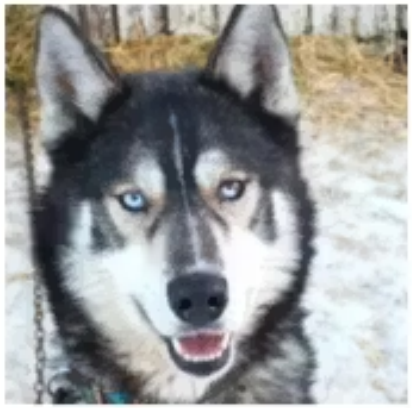
Don’t want to fit to spurious features

- There could be correlations between features in your train set and the test label that do not reflect a true causal relationship
- E.g. classifying wolves vs. dogs



Don’t want to fit to spurious features

- There could be correlations between features in your train set and the test label that do not reflect a true causal relationship
- E.g. classifying wolves vs. dogs



(a) Husky classified as wolf

(b) Explanation

Figure 11: Raw data and explanation of a bad model’s prediction in the “Husky vs Wolf” task.

	Before	After
Trusted the bad model	10 out of 27	3 out of 27
Snow as a potential feature	12 out of 27	25 out of 27

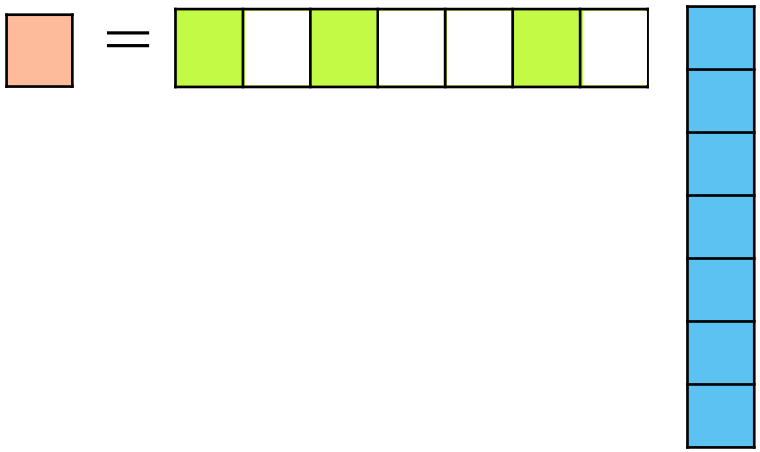
Table 2: “Husky vs Wolf” experiment results.

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

$$\widehat{y}_i = \widehat{w}_{\text{LS}}^T x_i$$



$$= \sum_{j=1}^{d} \widehat{w}_{\text{LS}}[j] \times x_i[j] = \sum_{j:w_{\text{LS}}[j]\neq 0} \widehat{w}_{\text{LS}}[j] \times x_i[j]$$

Computational complexity decreases from $2d$ to $2k$ for k-sparse $\widehat{w}_{\text{LS}}$

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	Dishwasher
Single Family	Garbage disposal
Year built	Microwave
Last sold price	Range / Oven
Last sale price/sqft	Refrigerator
Finished sqft	Washer
Unfinished sqft	Dryer
Finished basement sqft	Laundry location
# floors	Heating type
Flooring types	Jetted Tub
Parking type	Deck
Parking amount	Fenced Yard
Cooling	Lawn
Heating	Garden
Exterior materials	Sprinkler System
Roof type	
Structure style	

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

Finding best subset of features that explain the outcome/label: Exhaustive

- Try all subsets of size 1, 2, 3, … and one that minimizes validation error
  - Problem? # Exponential in d
  - Any Ideas?

Finding best subset: Greedy

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

Forward Greedy

1: $T \leftarrow \emptyset$

2: **For** $j = 1,\ldots,k$ **do**

3: $$j^* \leftarrow \arg\min_{\ell} \min_{w} \sum_{i=1}^{n} \left(y_i - \sum_{j \in T \cup \{\ell\}} w[j] \times x_i[j] \right)^2$$

4: $T \leftarrow T \cup \{j^*\}$

What feature can I add next to most reduce error, given I have these other features already?

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.

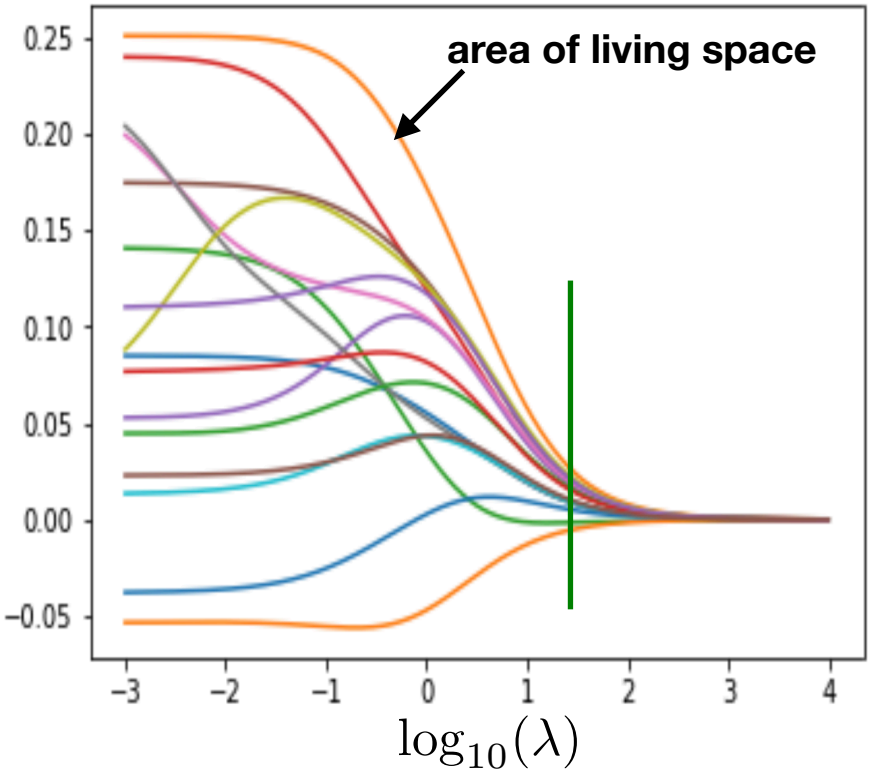
Computational complexity? **n^2**

Finding best subset: Regularize

Recall that Ridge regression makes coefficients small

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

w_i's



What do you notice for high value of lambda?

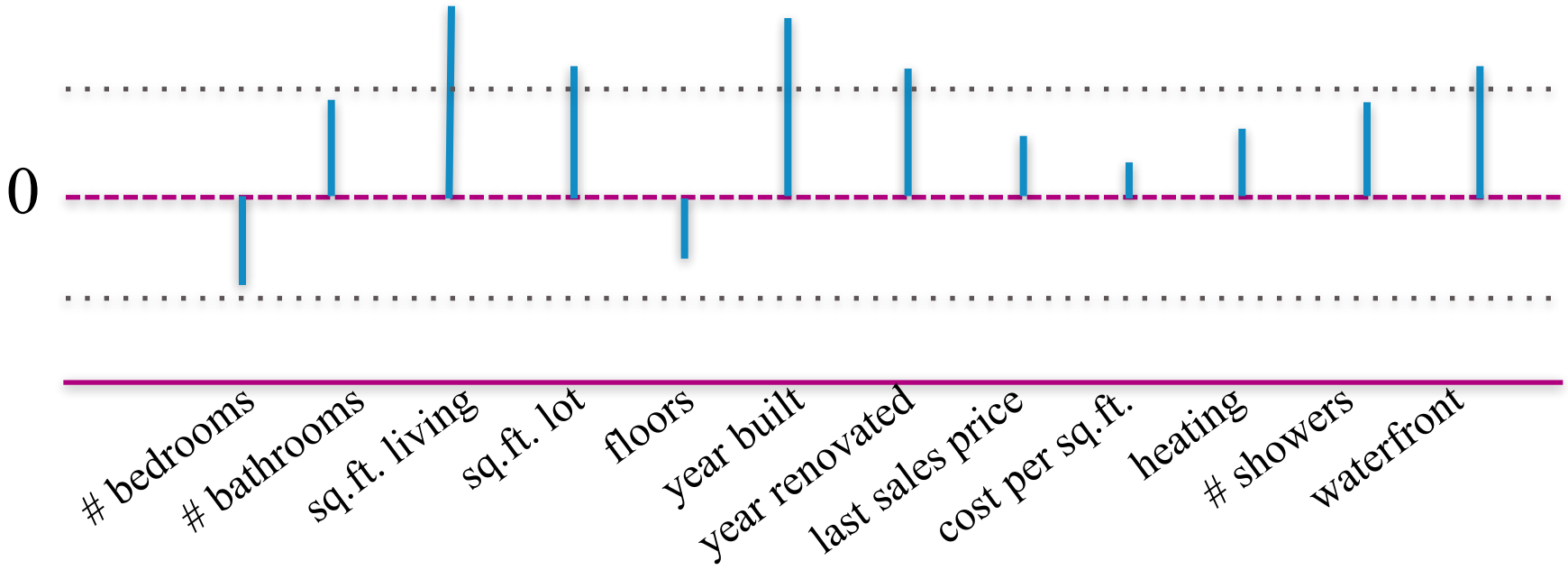
Most weights are non-zero. Not sparse

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?
  - Any issues?

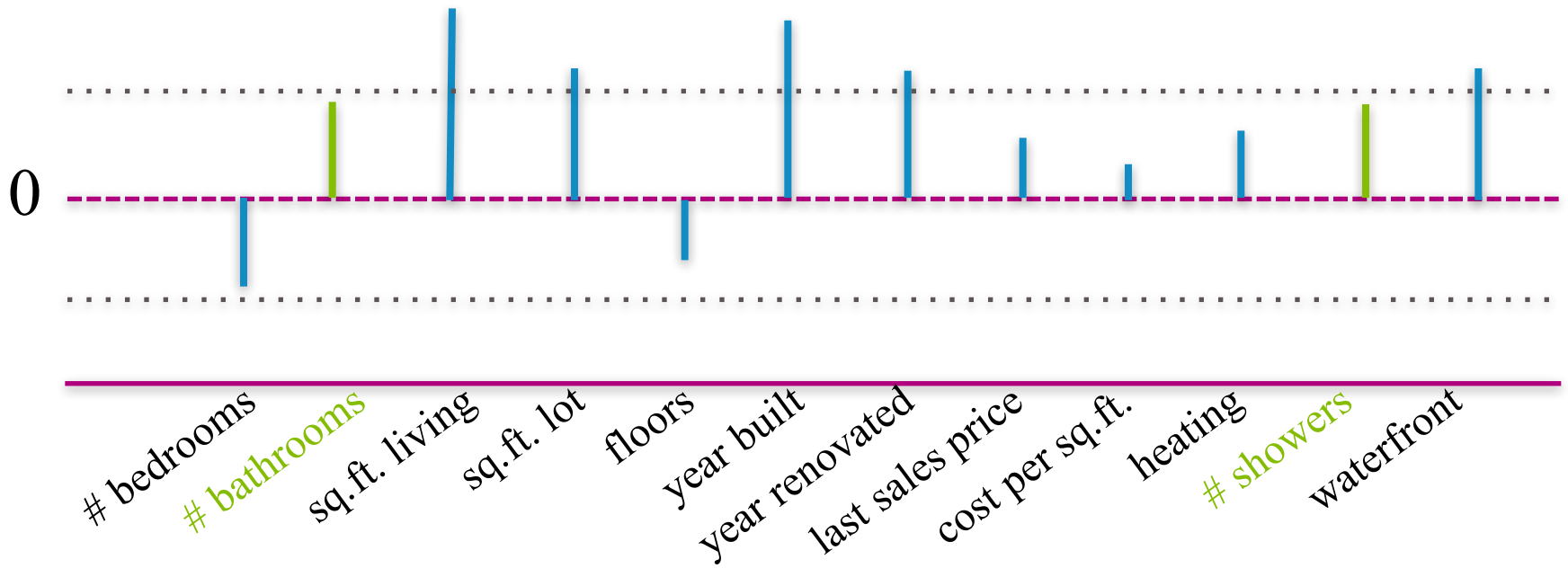
Doesn't analyze value of features together / as a set



Thresholded Ridge Regression

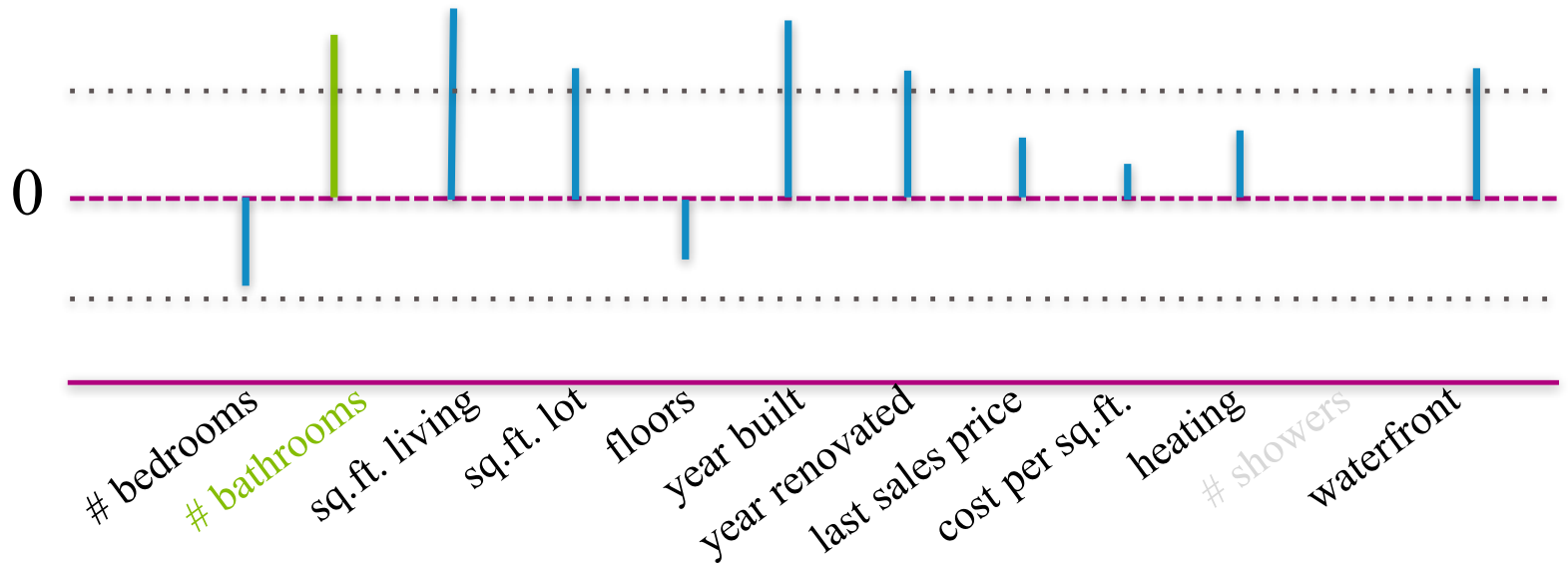
$$\widehat{w}_{ridge} = \arg\min_{w} \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

- Consider two related features (bathrooms, showers)
- Consider $w[\text{bath}] = 1$ and $w[\text{shower}] = 1$, and
  $w[\text{bath}] = 2$ and $w[\text{shower}] = 0$,
  which one does ridge regression choose? # w = [1,1]
  (assuming #bathroom=#showers in every house)



Thresholded Ridge Regression

- What if we **didn't** include showers? Weight on bathrooms increases!
- We want a feature selection scheme that selects one of (#bathroom) or (#showers) automatically, using the fact that if you delete #showers #bathroo is an important feature

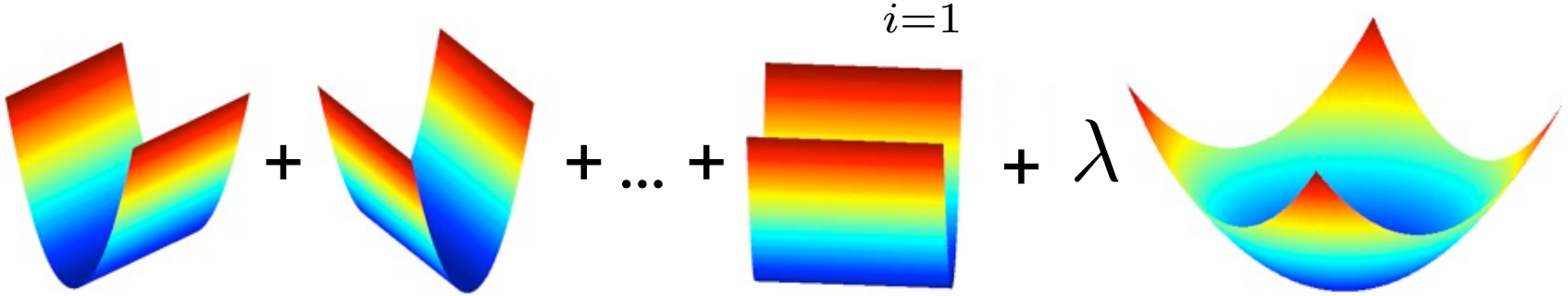


Can another regularizer perform selection automatically?

Ridge vs. Lasso Regression

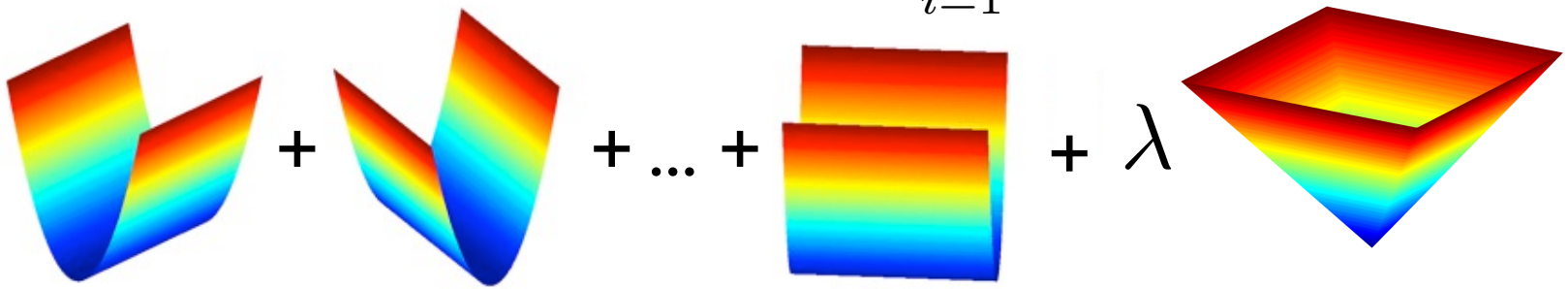
- Ridge Regression objective:

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



- Lasso objective:

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



Ridge vs. LASSO Regression

LASSO = Least Absolute Shrinkage and Selection Operator

- Recall Ridge Regression objective:

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

- sensitivity of a model w is measured in squared ℓ_2 norm $\|w\|_2^2$
- A principled method to get sparse model is **Lasso** with regularized objective:

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

- sensitivity of a model w is measured in ℓ_1 norm:

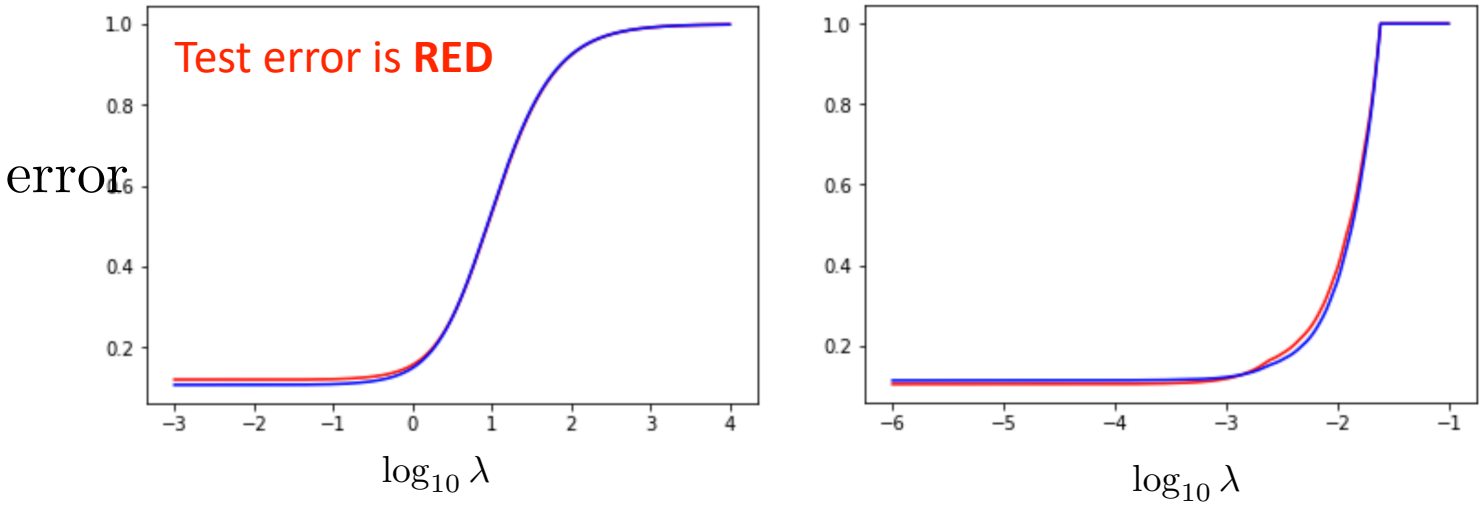
$$\|w\|_1 = \sum_{j=1}^{d} \left|w[j]\right|$$

absolute value

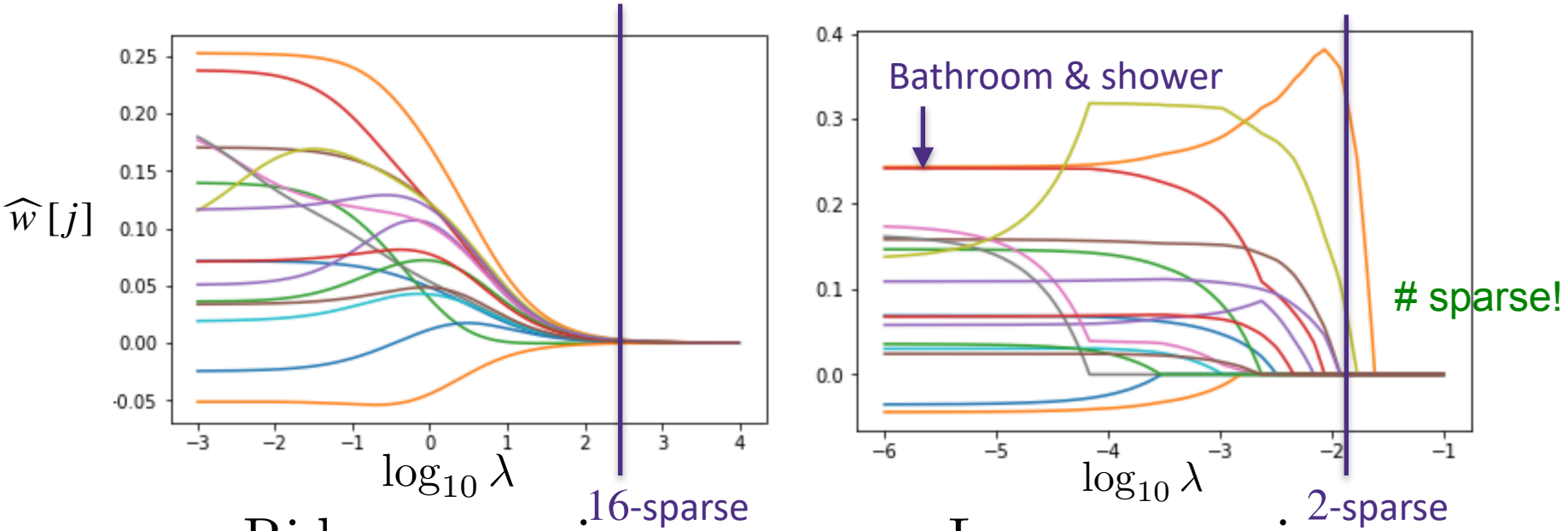
ℓ_p-norm of a vector $w \in \mathbb{R}^d$ is

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

Example: house price with 16 features



- **Regularization path for LASSO shows that weights drop to exactly zero as λ increases**



Ridge regression

Lasso regression

LASSO regression naturally gives sparse features

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

  why do we need to retrain?

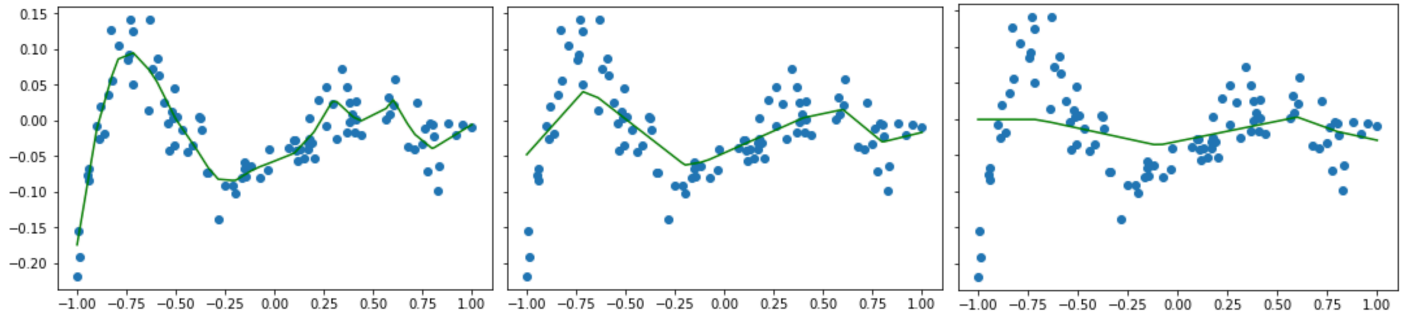
Example: piecewise-linear fit

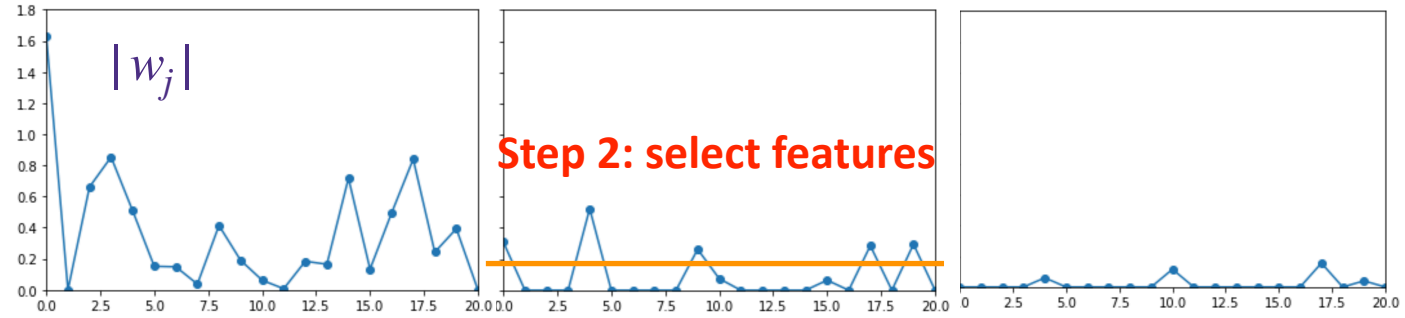
$$h_0(x) = 1$$
$$h_i(x) = [x + 1.1 - 0.1i]^+$$

- We use LASSO on the piece-wise linear example

Step 1: find optimal λ^*

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





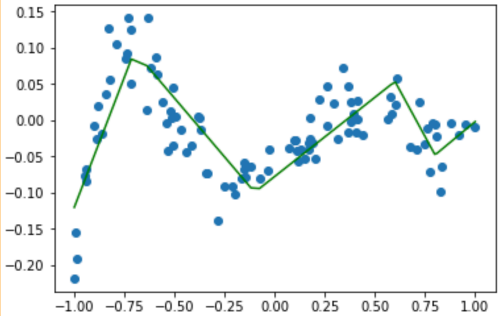
Step 2: select features

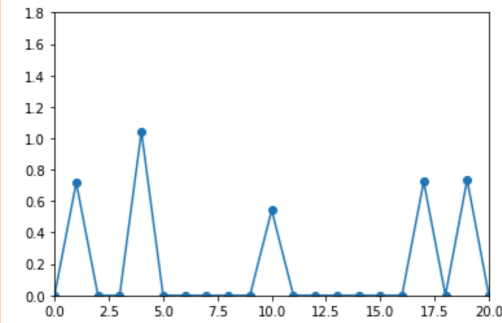
$\lambda = 10^{-8}$ $\lambda = 10^{-4}$ $\lambda = 2 \times 10^{-4}$

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

Step 3: retrain

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





$\lambda = 0$

but only use selected features

Still not clear?

“Sensitivity to the features is okay, as long as you have the right features”

— A past 446 student

Regularized Least Squares

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

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

Theorem:

λ-regularized LS ⟷ μ-constrained LS

$$\min_w \|Y - Xw\|_2^2 + \lambda \|w\|_2^2$$

(Where big weights are penalized)

$$\min_w \|Y - Xw\|_2^2 \quad \text{s.t.} \quad \|w\|_2^2 \leq \mu$$

(Where weights must stay smaller than a constraint value μ)

Regularized Least Squares

- Regularized optimization:

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

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

- For any $\lambda^* \geq 0$ for which $\hat{w}_r$ achieves the minimum, there exists a $\mu^* \geq 0$ such that the solution of the constrained optimization, $\widehat{w}_c$, is the same as the solution of the regularized optimization , $\widehat{w}_r$, where

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

- so there are pairs of (λ, μ) whose optimal solution $\widehat{w}_r$ are the same for the regularizes optimization and constrained optimization

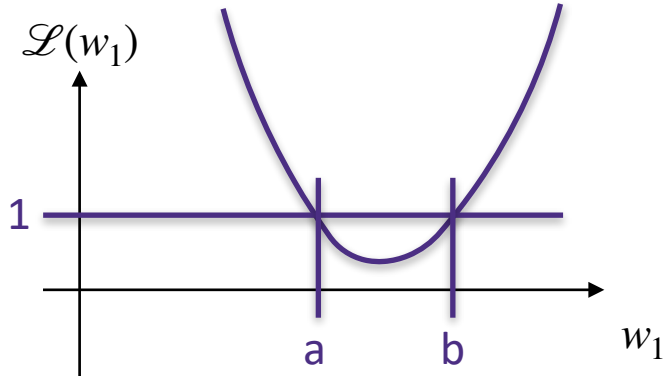
Why does LASSO give sparse solutions?

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

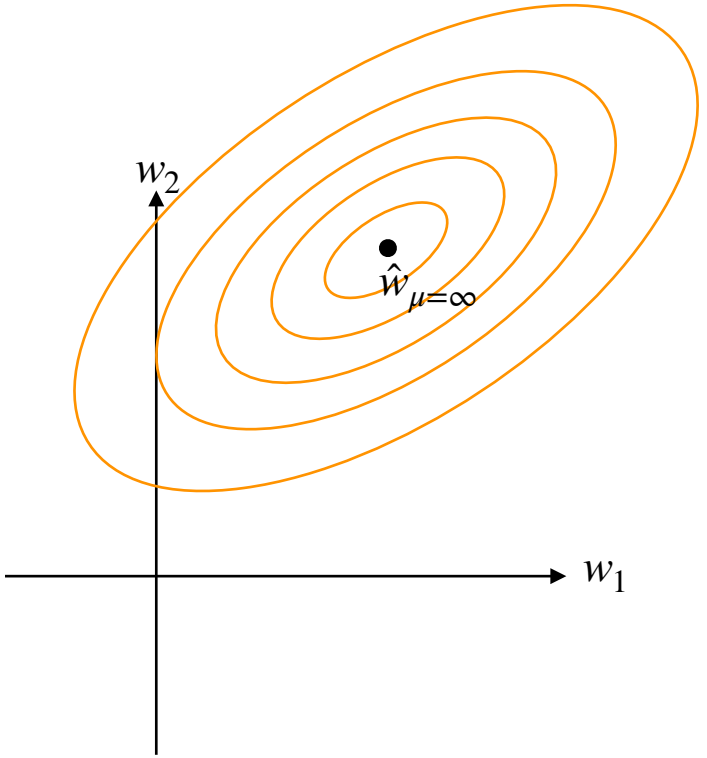
$$\text{subject to} \quad \|w\|_1 \leq \mu$$

- 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 $\hat{w}_{\mu=\infty} = \hat{w}_{\text{LS}}$

1-D example with quadratic loss



Level set of $\mathscr{L}(w_1)$ at value 1 is {a,b}



Why does Lasso give sparse solutions?

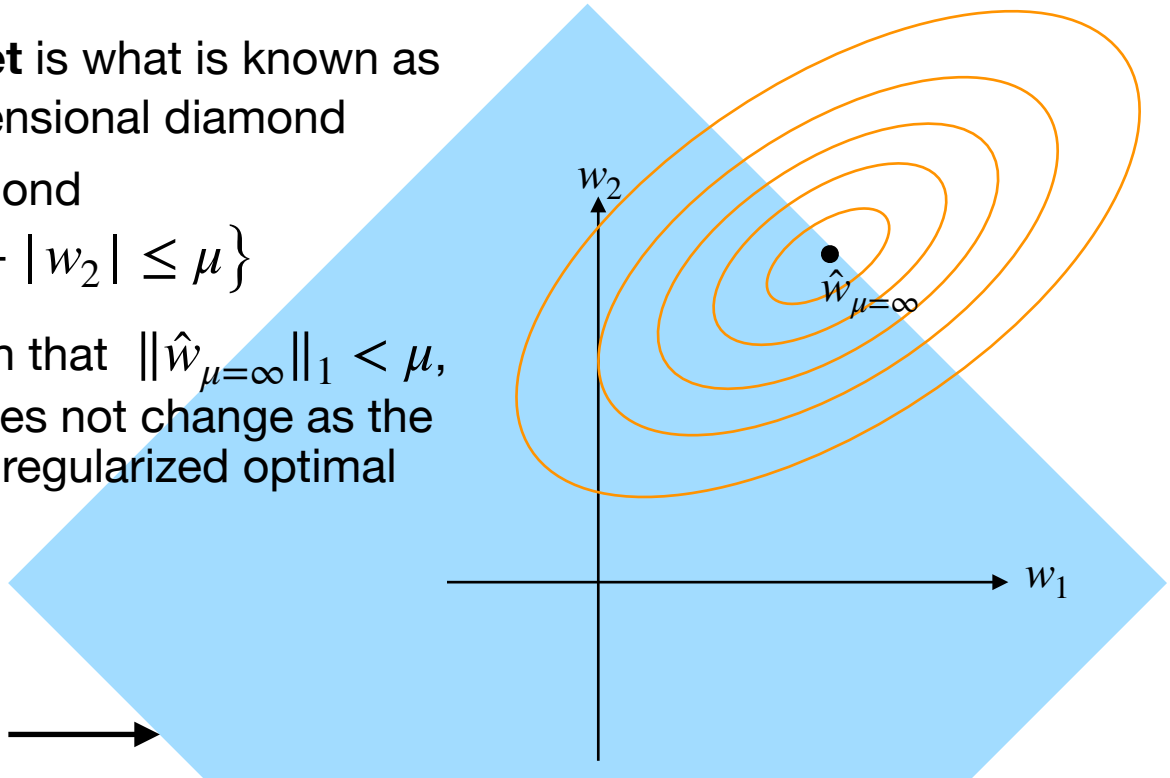
$$\text{minimize}_w \quad \sum_{i=1}^{n} (w^T x_i - y_i)^2$$

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

- 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

$$\{(w_1, w_2) \mid |w_1| + |w_2| \leq \mu\}$$

- 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



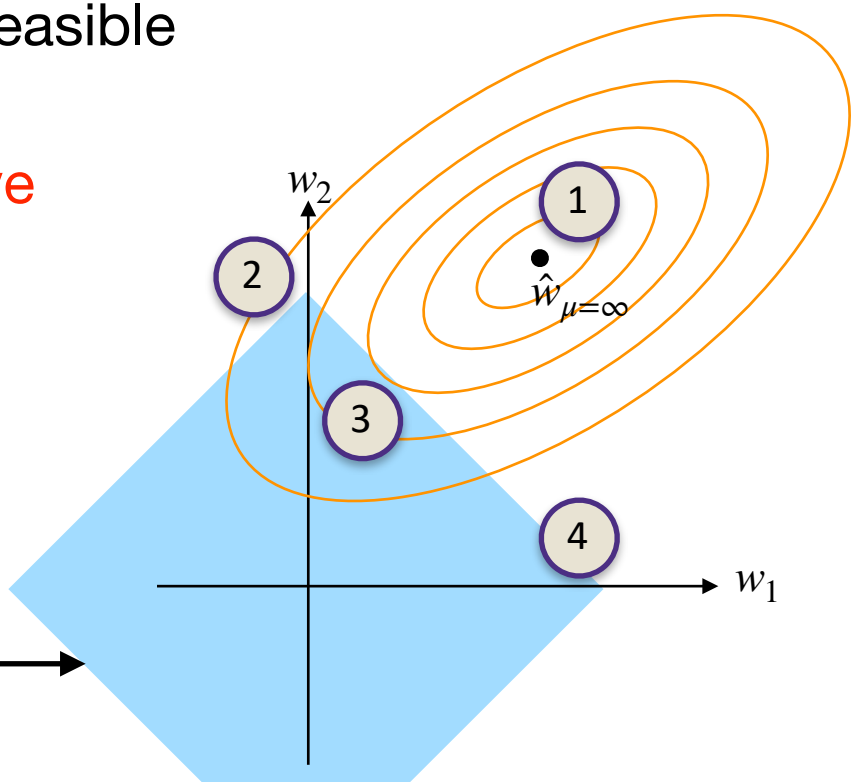
feasible set: $\{w \in \mathbb{R}^2 \mid \|w\|_1 \leq \mu\}$ ⟶

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 λ) the feasible set (blue diamond) shrinks
- The optimal solution of the above optimization is ?



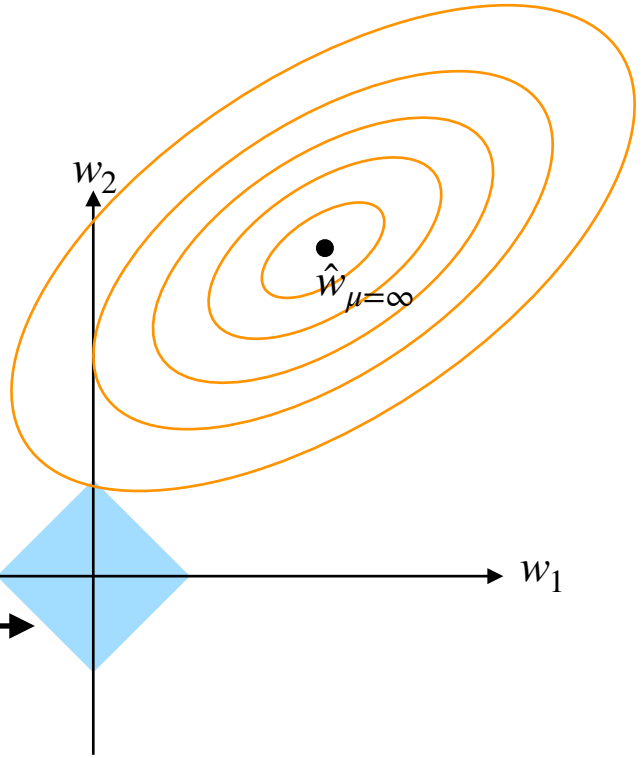
feasible set: $\{w \in \mathbb{R}^2 \mid \|w\|_1 \leq \mu\}$

Why does Lasso give sparse solutions?

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

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

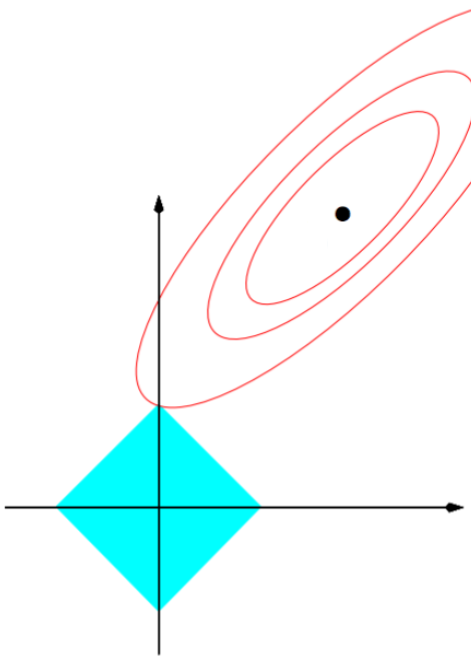
- For small enough μ, the optimal solution becomes **sparse**
- This is because the L_1-ball is "pointy",i.e., has sharp edges aligned with the axes

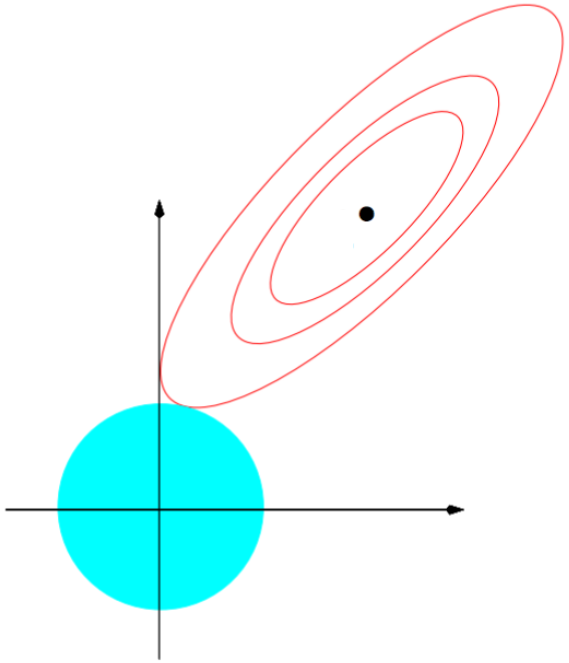


feasible set: $\{w \in \mathbb{R}^2 \mid \|w\|_1 \leq \mu\}$

Constrained Least Squares

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





$$\text{minimize}_w \sum_{i=1}^{n} (w^T x_i - y_i)^2 \quad \text{subject to} \quad \|w\|_1 \leq \mu$$

$$\text{minimize}_w \sum_{i=1}^{n} (w^T x_i - y_i)^2 \quad \text{subject to} \quad \|w\|_2^2 \leq \mu$$

L1 Ball in Higher Dimensions

> L1 ball 3 dimensions