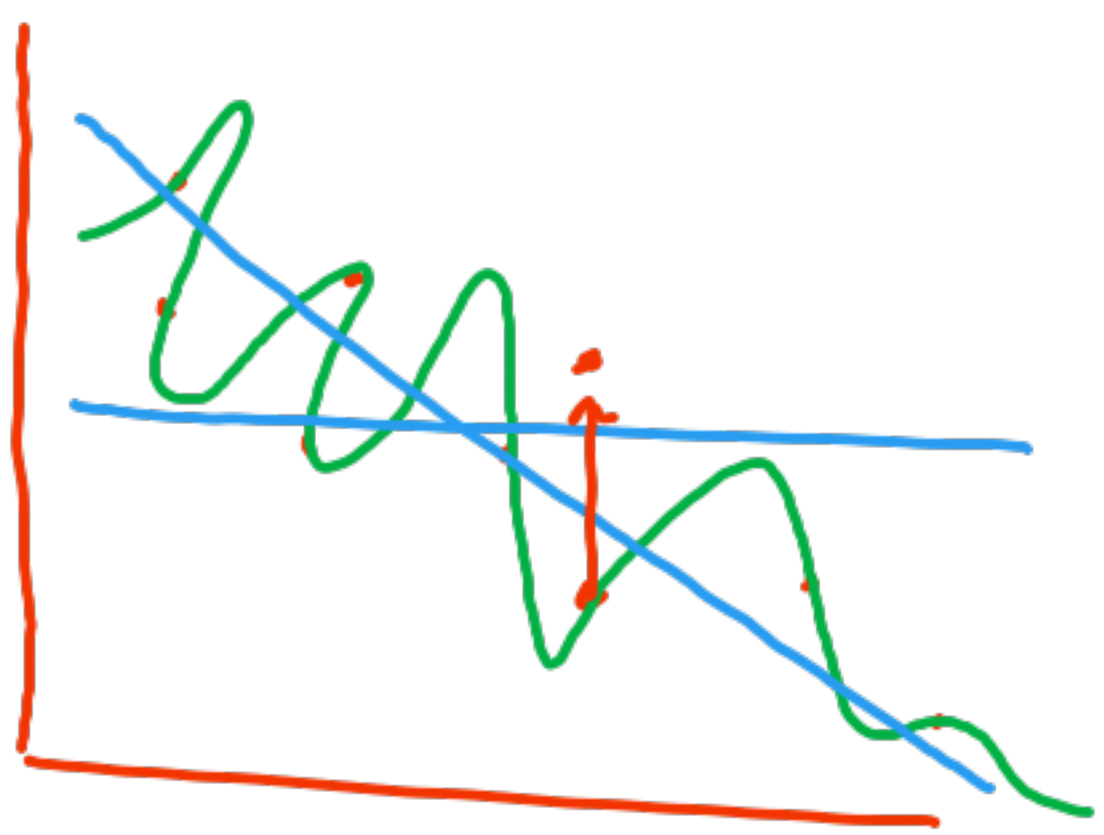
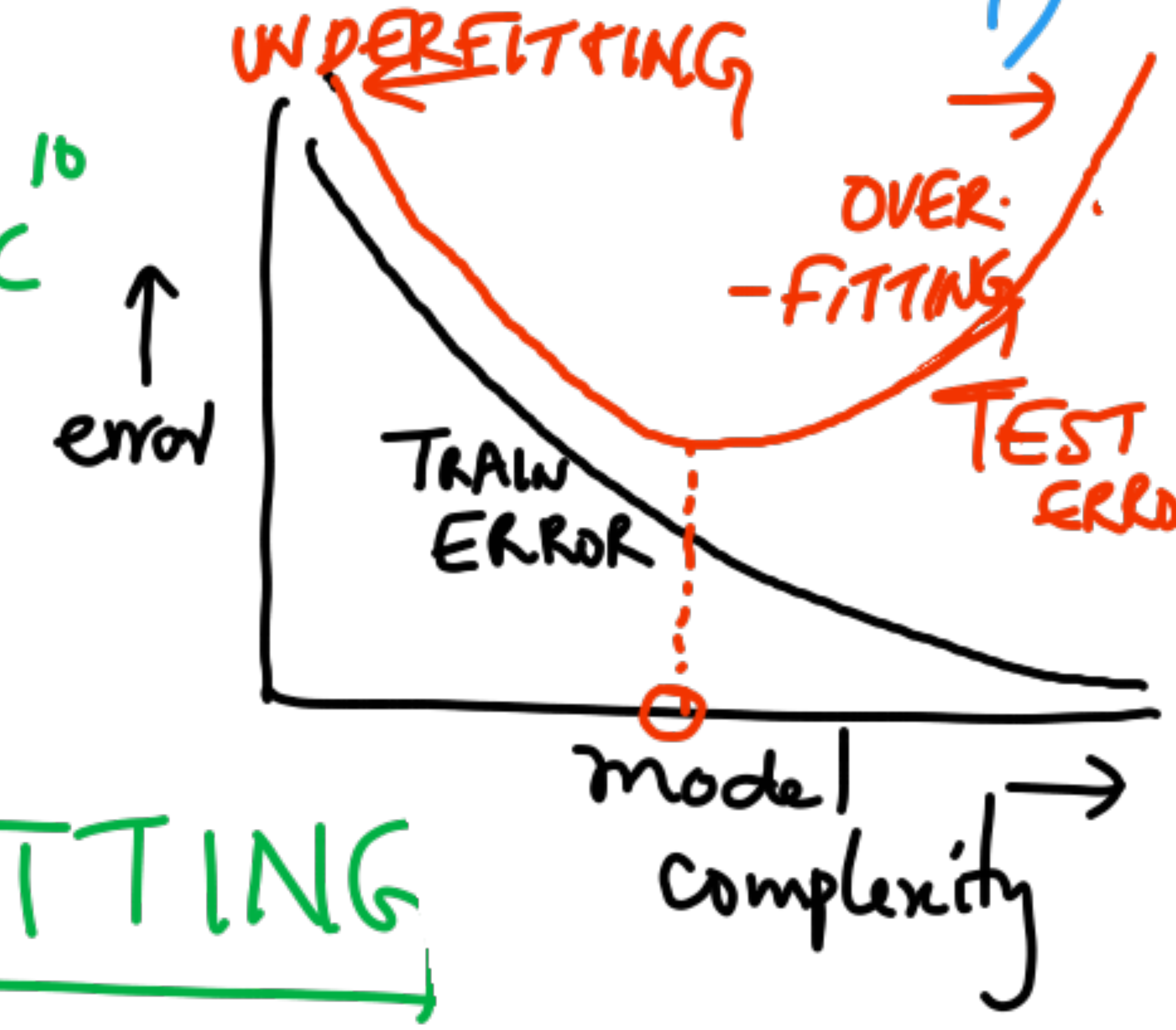
12/8/24 CS 725



$$h_w(x) = w_0 + w_1 x \quad \text{(UNDERFITTING)}$$

$$h_w(x) = w_0 + w_1 x + w_2 x^2 + \dots w_{10} x^{10}$$



UNDERFITTING

- Training error is high
- Model is overly simple (low capacity, low expressitivity)
- Not estimating the input-output mapping well

OVERFITTING

- Training error is nil or very smal, but val/test error is high
- The variance across predictor functions is high
- Model is overly complex

How to combat overfitting?

Tune the model (say degree of a polynomial) on a val set to identify a good fit

Cons: EXPENSIVE

Principled way to curb overfitting: REGULARIZATION

Combining model fit with penalizing overly complex models

For LR, optimize regularized loss function:

$$L_{reg}(w) = L_{LSE}(w, \mathcal{D}_{train}) + \lambda \underline{R(w)}$$

REGULARIZER FUNCTION

$$L_{reg}(w) = \underbrace{L_{MSE}(w, \mathcal{D}_{train})}_{\text{measure of fit}} + \underbrace{\lambda R(w)}_{\text{measure of model complexity}}$$

$R(w)$: Penalty function on w that constrains the values of w

Such functions are called "shrinkage" functions

Two popular regularizers for linear regression are:

1. L_2-regularized LR (RIDGE REGRESSION)
2. L_1-regularized LR (LASSO REGRESSION)
   → Least Absolute Shrinkage & Selection Operator

1 RIDGE REGRESSION

regularization Scaling

$$w_{RIDGE} = \underset{w}{\operatorname{argmin}} \underbrace{\| y - \phi w \|_2^2 + \lambda \| w \|_2^2}_{\mathcal{L}_{ridge}}$$

$$\nabla_w \mathcal{L}_{ridge} = 0 \Rightarrow -2\phi^T (y - \phi w) + 2\lambda w = 0$$

$$\Rightarrow \boxed{w_{ridge} = \left(\phi^T \phi + \lambda I \right)^{-1} \phi^T y}$$

$$W_{ridge} = (\phi^T\phi + \lambda I)^{-1}\phi^T y$$

If $\lambda > 0$, then $(\phi^T\phi + \lambda I)$ is guaranteed to be invertible

CLAIM If $X \in \mathbb{R}_{n \times n}$ is positive definite, then X is invertible

A matrix X is positive definite iff for any non-zero vector v, $v^T X v > 0$.

If $v^T X v > 0 \Rightarrow Xv \neq 0 \Rightarrow X$ is invertible

Show that $\phi^T\phi + \lambda I$ is positive definite

$$v^T(\phi^T\phi + \lambda I)v > 0$$

$$\Rightarrow v^T\phi^T\phi v + \lambda v^T I v$$

$$\Rightarrow (\phi v)^T\phi v + \lambda v^T v$$

$$\Rightarrow \|\phi v\|_2^2 + \lambda \|v\|_2^2 > 0 \text{ if } \lambda > 0$$

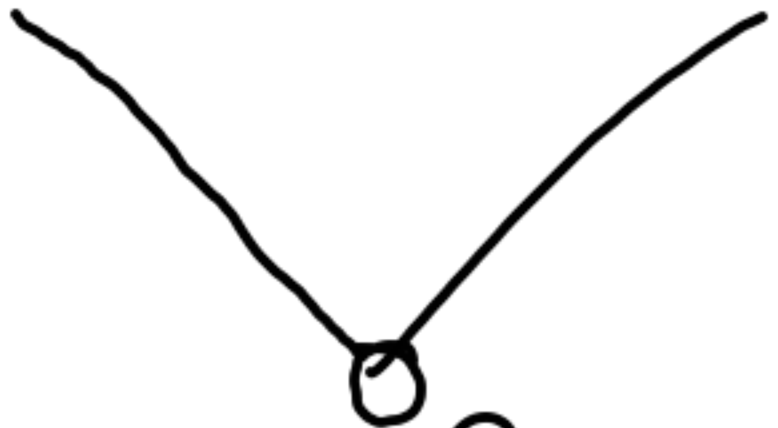
② LASSO REGRESSION

$$w_{LASSO} = \underset{w}{\operatorname{argmin}} \|y - \Phi w\|_2^2 + \lambda \|w\|_1$$

L_1-regularized LR does not have a closed form solution

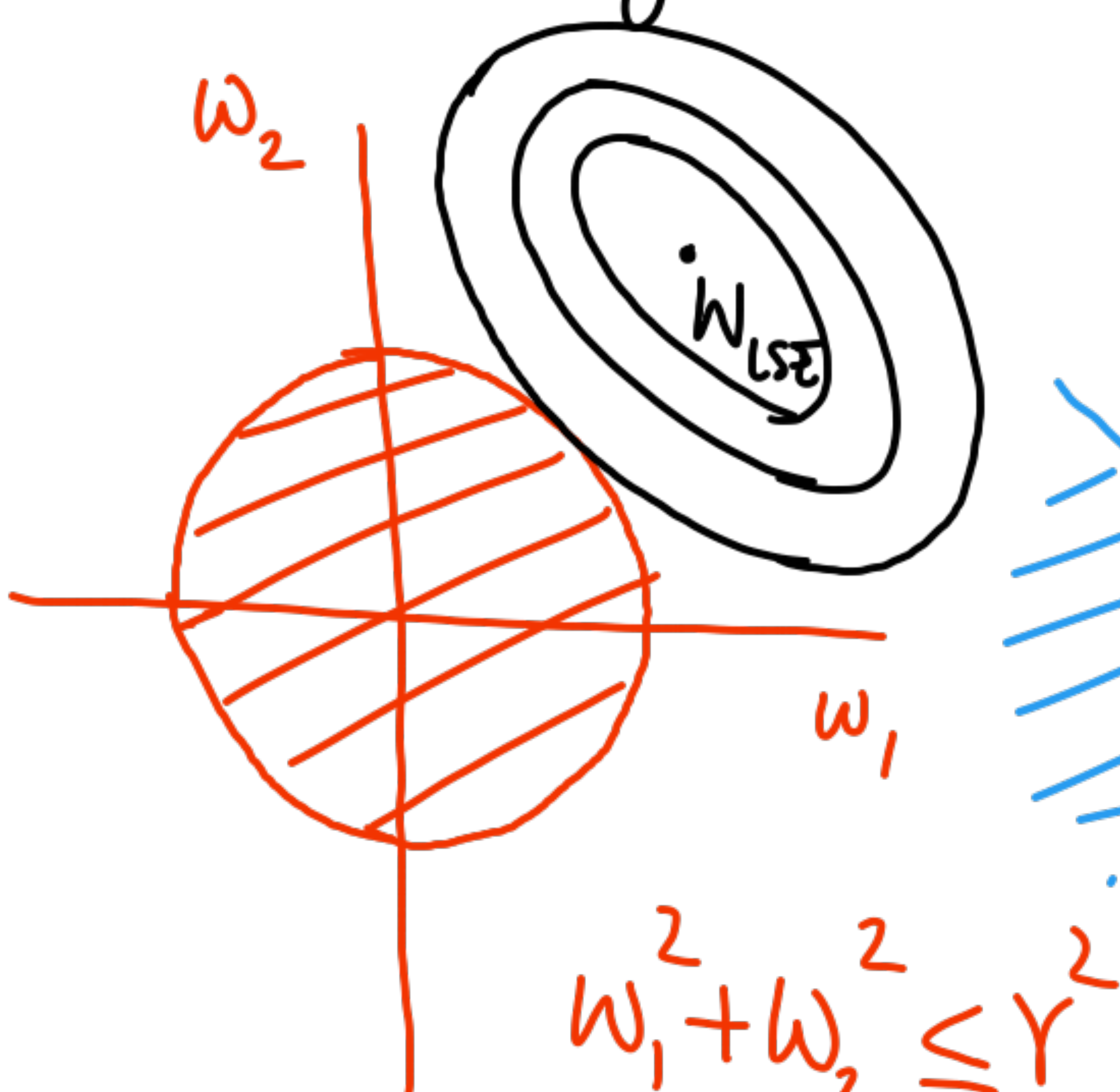
Solve for w_{LASSO}:

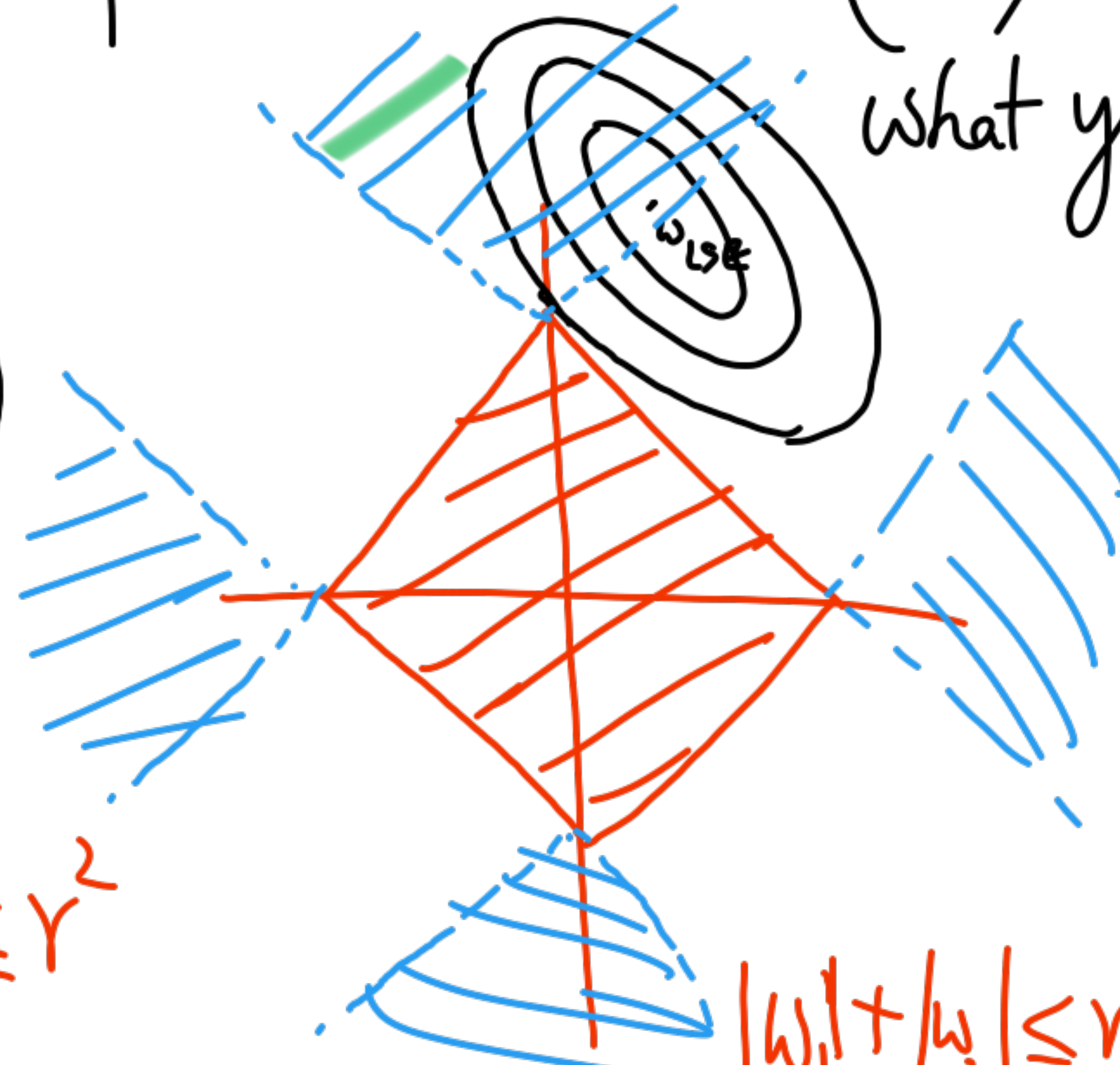
1. Quadratic Programming
2. Gradient descent using (Subgradients)



RIDGE (L_2-regularized) vs LASSO (L_1-regularized)

- LASSO yields sparser solutions (i.e., w is sparser than what you'd get with w_{ridge})





NOTE:

$$\underset{w}{\operatorname{argmin}} \|y-\phi w\|_2^2 + \lambda \|w\|_2^2$$

$$\Updownarrow$$

$$\underset{w}{\operatorname{argmin}} \|y-\phi w\|_2^2$$

$$\text{s.t. } \|w\|_2^2 \le t^2$$

L_1-regularized (LASSO) LR yields sparser solutions thus enabling FEATURE SELECTION

Can we combine L_2 & L_1? ELASTIC Regularization

Recall MLE; finding w that maximizes the likelihood of the observed data.

$$w_{MLE} = \underset{w}{\operatorname{argmax}} \log P(\mathcal{D} \mid w)$$

In MLE, the observations are random variables but not the parameters w.
What if we treat w as a random variable also, and define a prior on it?

The prior $P(w)$ encodes some prior beliefs about the problem

This results in a new parameter estimation technique called

Maximum Aposteriori (MAP) Estimation

MAP estimate ; $w_{MAP} = \underset{w}{\operatorname{argmax}}\, P(w \mid \mathcal{D})$] POSTERIOR Probability of w given $\mathcal{D}$

$$= \underset{w}{\operatorname{argmax}}\, \underbrace{P(\mathcal{D} \mid w)}_{\text{LIKELIHOOD}} \underbrace{P(w)}_{\text{PRIOR}}$$

Back to the coin toss problem with θ being the probability of landing on heads. $\mathcal{D}$ = N cointosses, N_H heads, N_T tails

$$P(\mathcal{D}|\theta) = \theta^{N_H}(1-\theta)^{N_T}$$

$$P(\theta) = ?$$

CANDIDATE for the prior: BETA DISTRIBUTION

$$\mathrm{Beta}(\theta;\alpha,\beta) = \frac{1}{c}\theta^{\alpha-1}(1-\theta)^{\beta-1}$$