CSE 446
Gradient Descent

Natasha Jaques

UNIVERSITY of WASHINGTON

How do we find optimal weights?

- This is related some questions you might have so far in this course
  - Why do we use quadratic loss, $\sum_{i=1}^{n} (y_i - w^T x_i)^2$?
  - Why is Gaussian noise so popular?

$$f(z) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\frac{(z-\mu)^2}{\sigma^2}}$$

  - Why was Ridge Regression with L_2 regularizer, $\|w\|_2^2$, the first to be used?
  - When we want sparsity, why do we use L_1 regularizer, $\|w\|_1$, and not $L_{0.5}$ regularizer, $\|w\|_{0.5}$?

Convexity

local minima is the global minimum

Why gradient descent?

- Standard ML paradigm: Define loss, then optimize

$$\widehat{w}_{LS} = \arg\min_{w} ||\mathbf{y} - \mathbf{X}w||_2^2$$

$\rightarrow 2X^T(Xw - y) = 0$

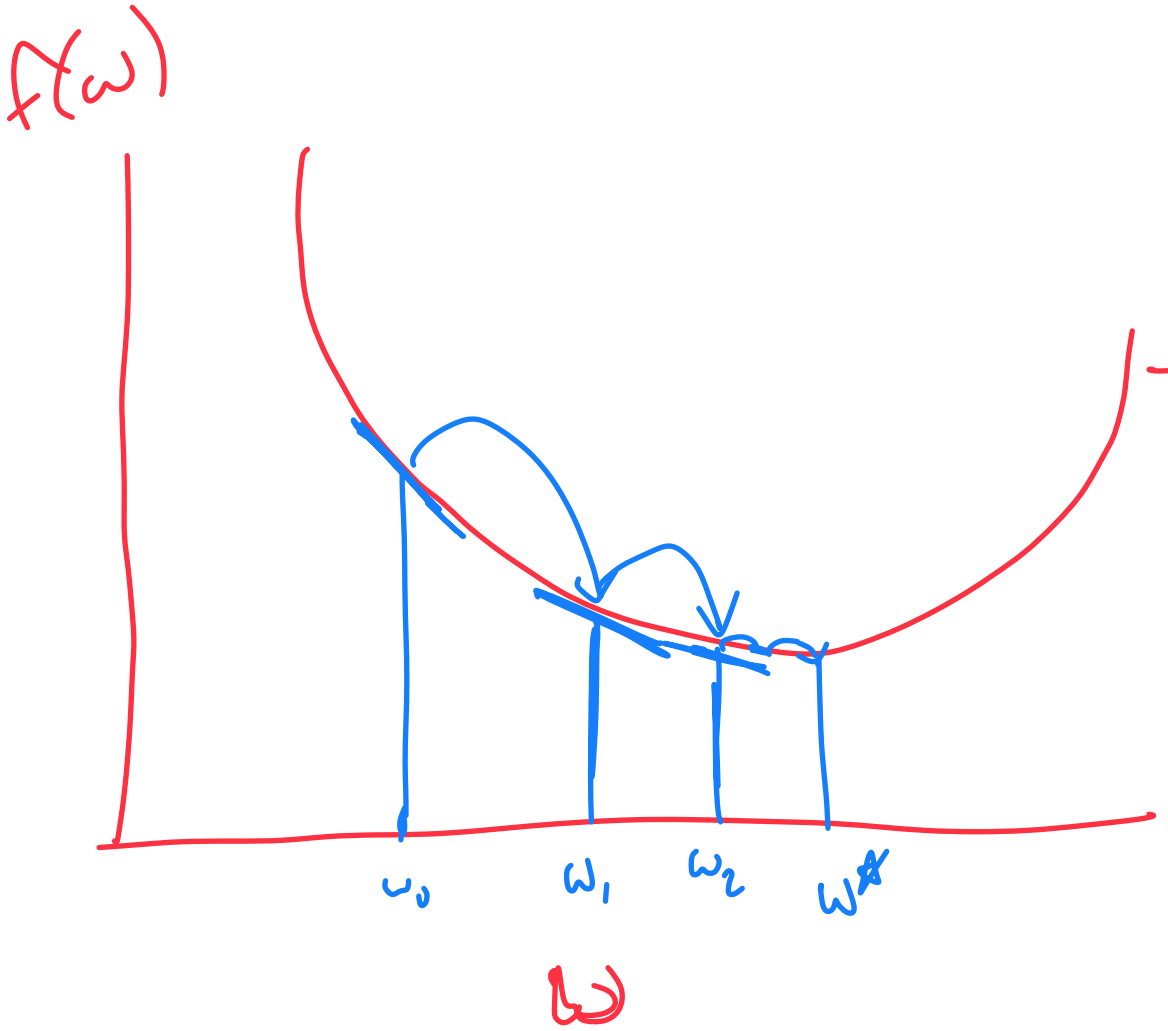
$\hat{w} = (X^TX)^{-1}X^Ty$

$+ \lambda ||w||_1$; no closed form solution ?

- But, no closed-form solutions for most losses we use in practice.
- Key idea: Iterative methods
- Used everywhere!

→ start with a guess for w
→ iteratively refine to minimize loss

Gradient descent in one dimension

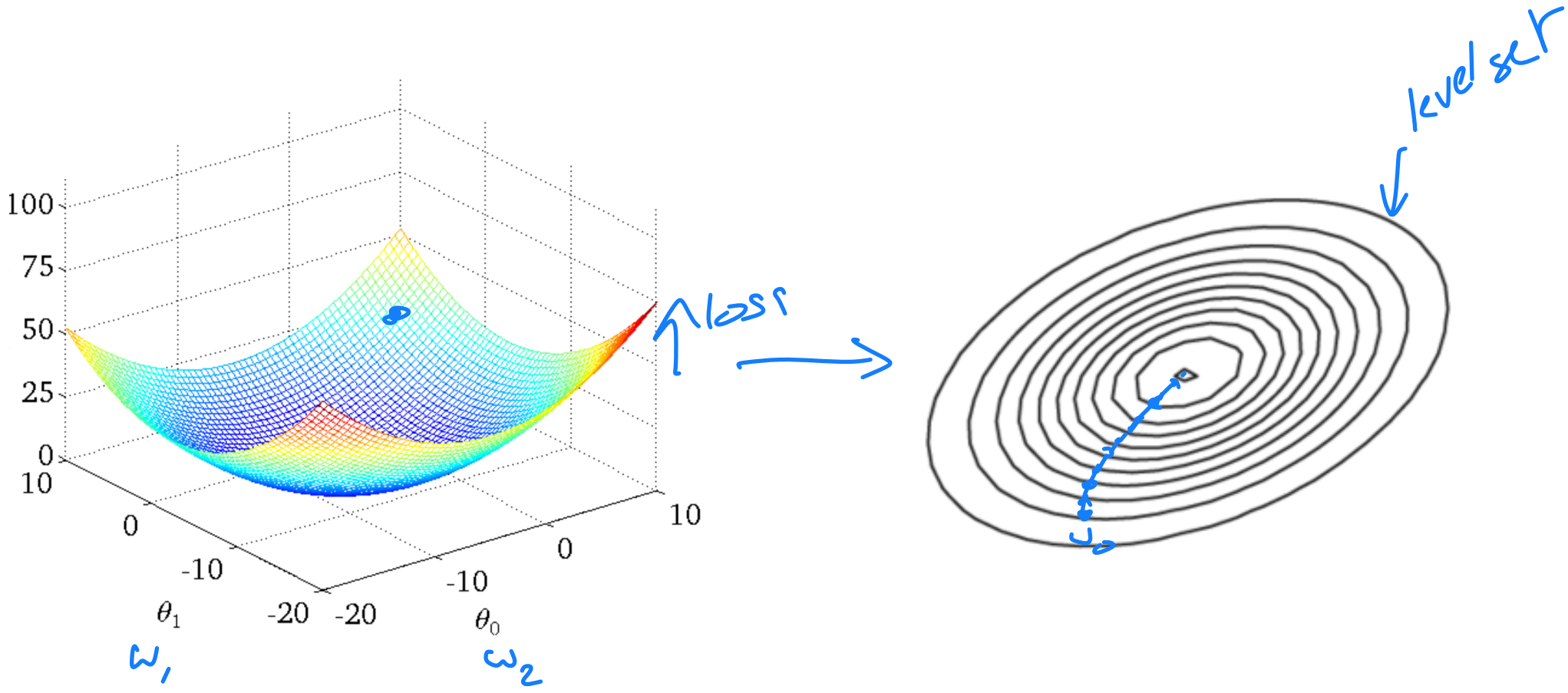


Step direction: –gradient

Step size: ∝ |gradient|

η × |gradient|

Gradient descent in multiple dimensions



Lecture plan

- Gradient descent algorithm + examples

- Theoretical analysis
  - When does it work?
  - How quickly does it converge?
  - How do we choose a step size?
  - Key idea: Convexity

- Not tested on proof details, but concepts are important & practical

Algorithm: Gradient descent

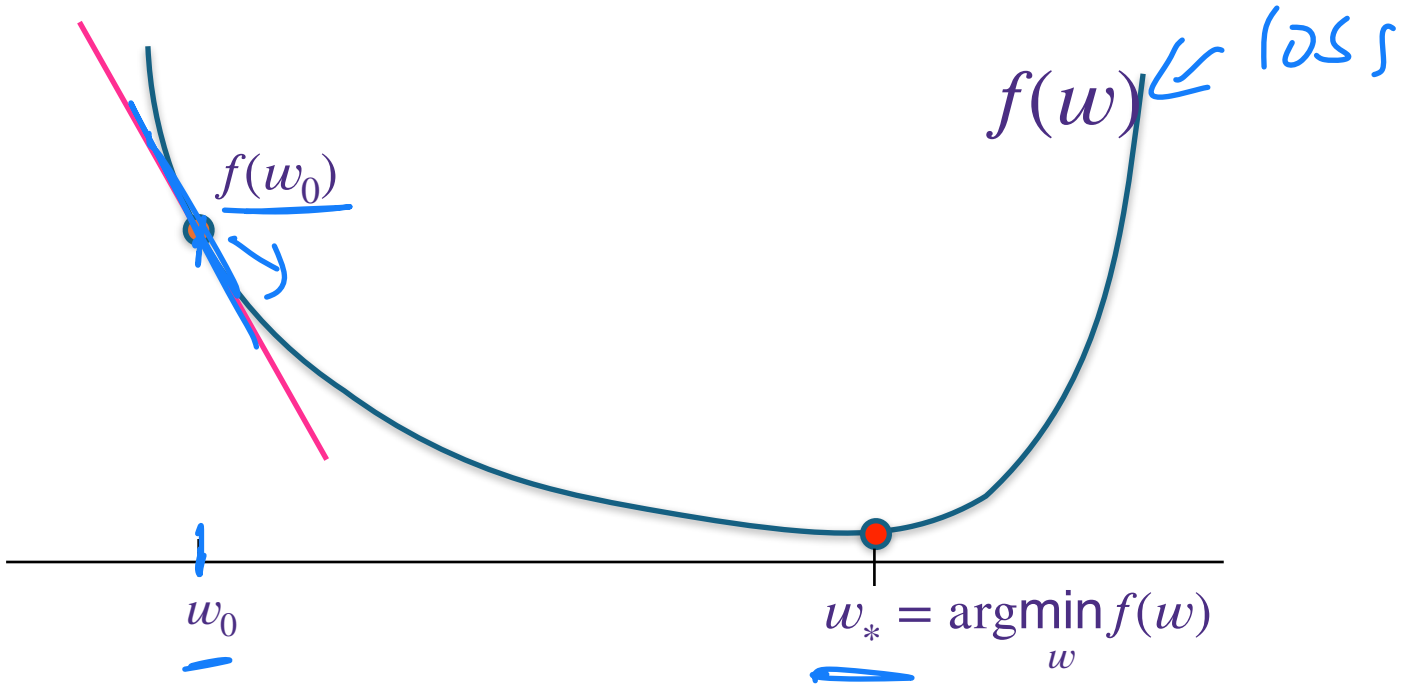
Algorithm

For t=0,1,2,3, …

$$w_{t+1} = w_t - \eta \frac{df(w)}{dw}\Big|_{w=w_t}$$

Hyperparameters:

- Initial point w_0
- Step size η



Algorithm: Gradient descent

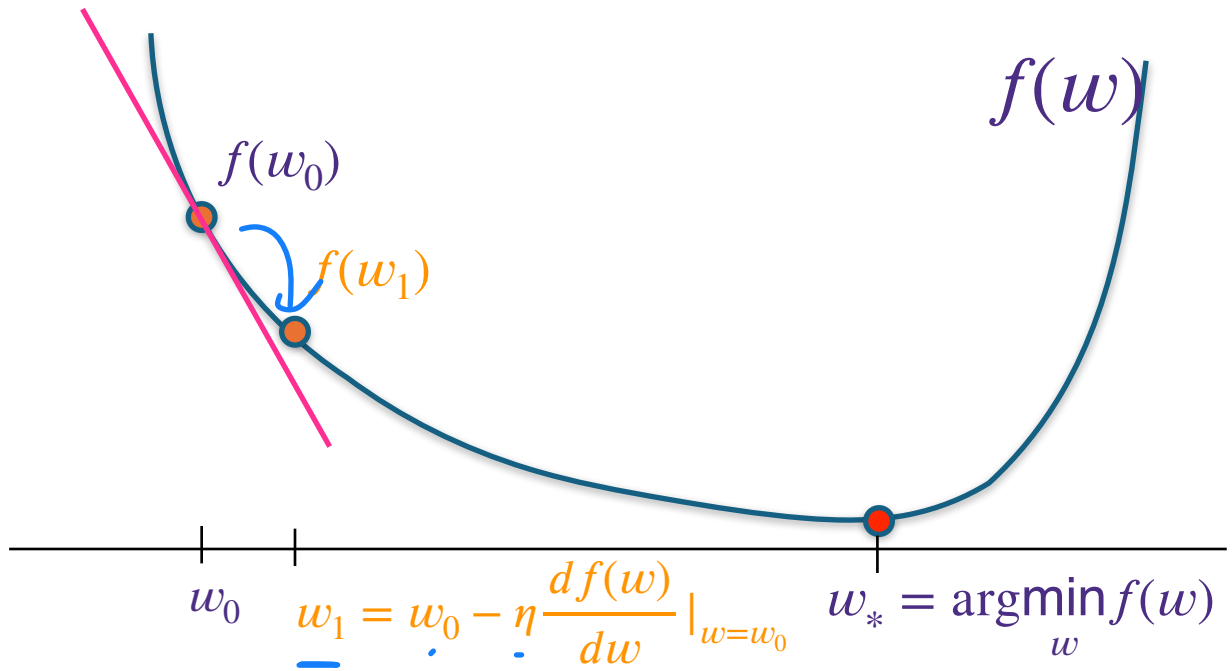
Algorithm

For t=0,1,2,3, …

$$w_{t+1} = w_t - \eta \frac{df(w)}{dw}|_{w=w_t}$$

Hyperparameters:

- Initial point w_0
- Step size η

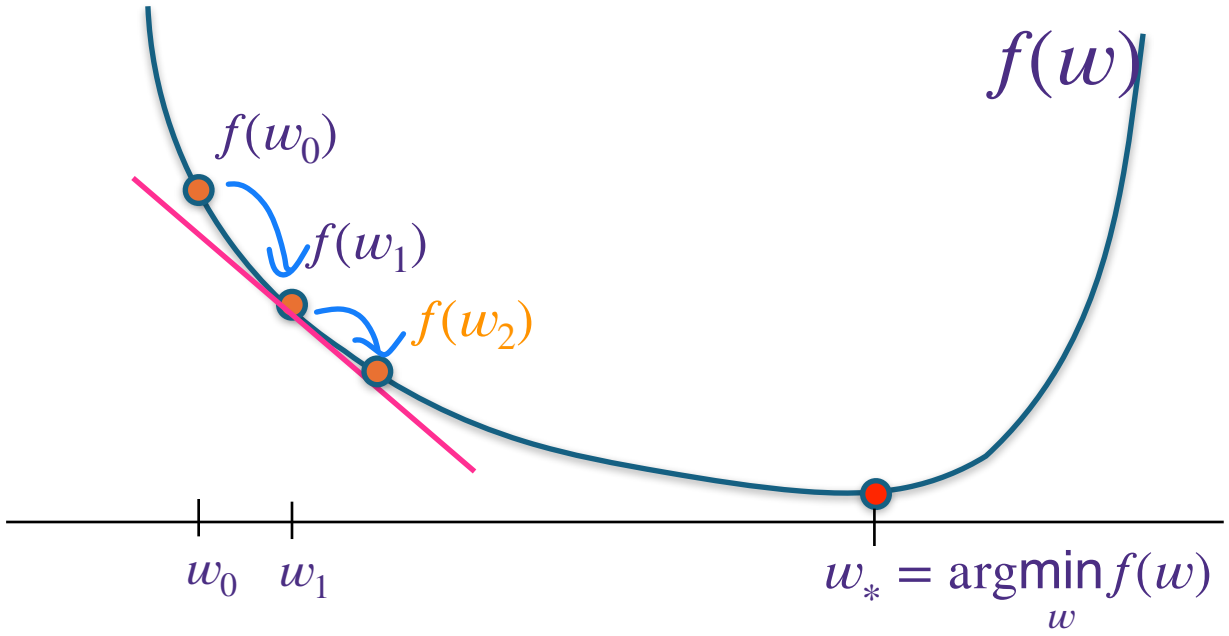


Algorithm: Gradient descent

Algorithm
For t=0,1,2,3, …

$$w_{t+1} = w_t - \eta \frac{df(w)}{dw}\Big|_{w=w_t}$$

Hyperparameters:
- Initial point w_0
- Step size η

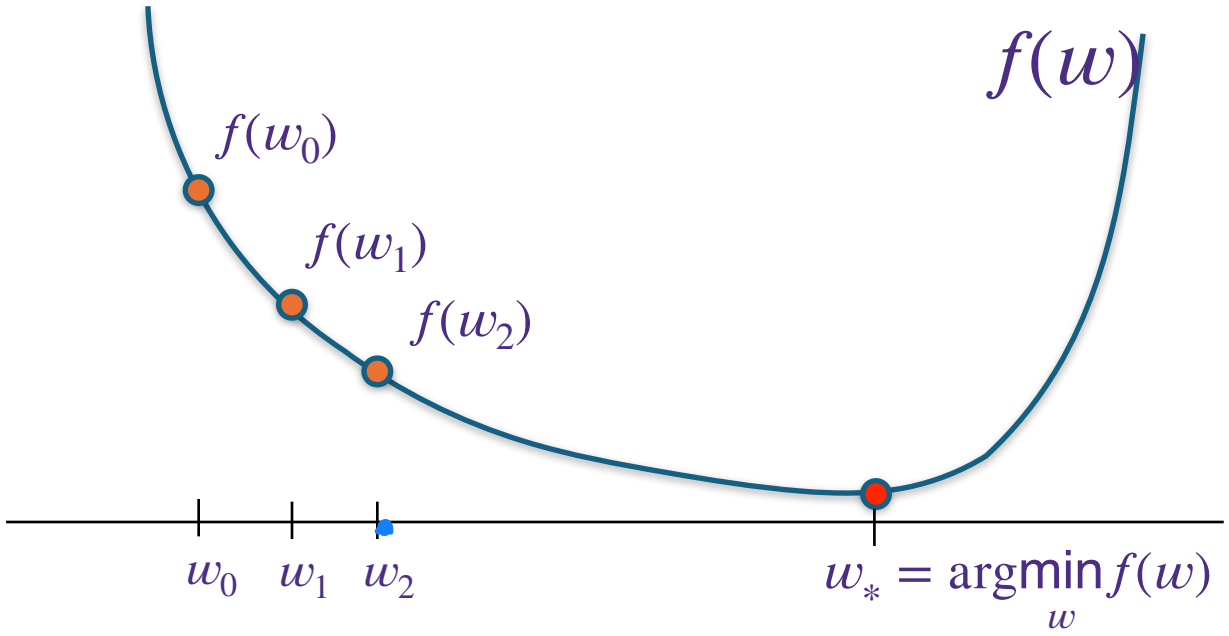


Algorithm: Gradient descent

Algorithm
For t=0,1,2,3, …

$$w_{t+1} = w_t - \eta \frac{df(w)}{dw}\Big|_{w=w_t}$$

Hyperparameters:
- Initial point w_0
- Step size η

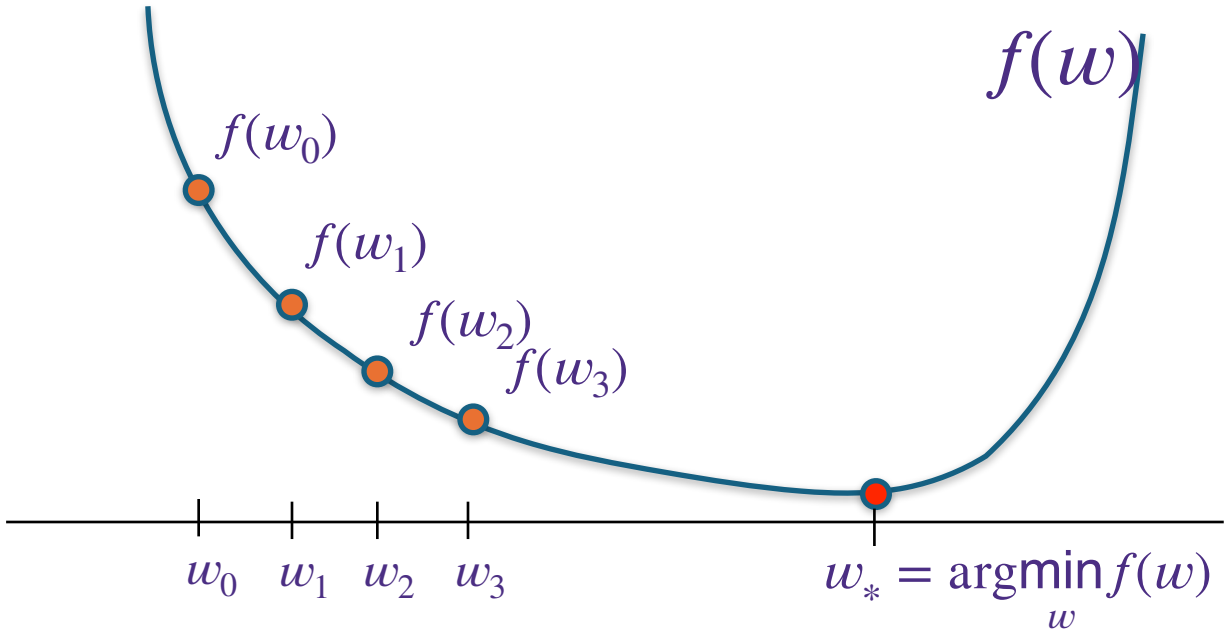


Algorithm: Gradient descent

Algorithm
For t=0,1,2,3, …

$$w_{t+1} = w_t - \eta \frac{df(w)}{dw}\Big|_{w=w_t}$$

Hyperparameters:
- Initial point w_0
- Step size η

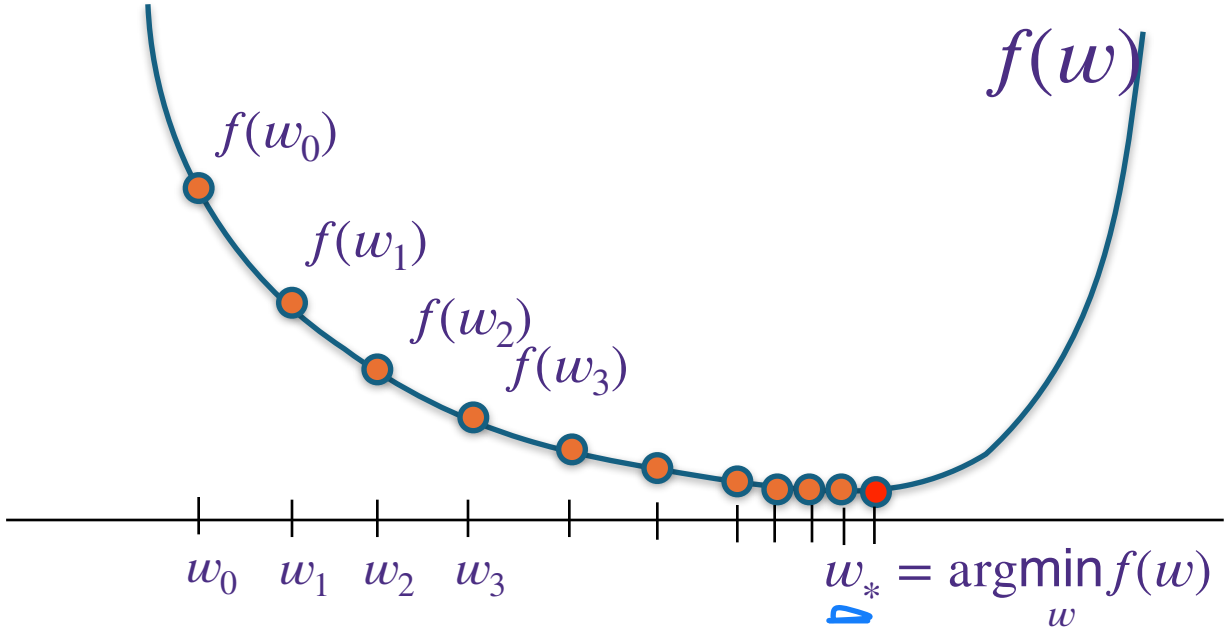


Algorithm: Gradient descent

Algorithm
For t=0,1,2,3, …

$$w_{t+1} = w_t - \eta \frac{df(w)}{dw}\Big|_{w=w_t}$$

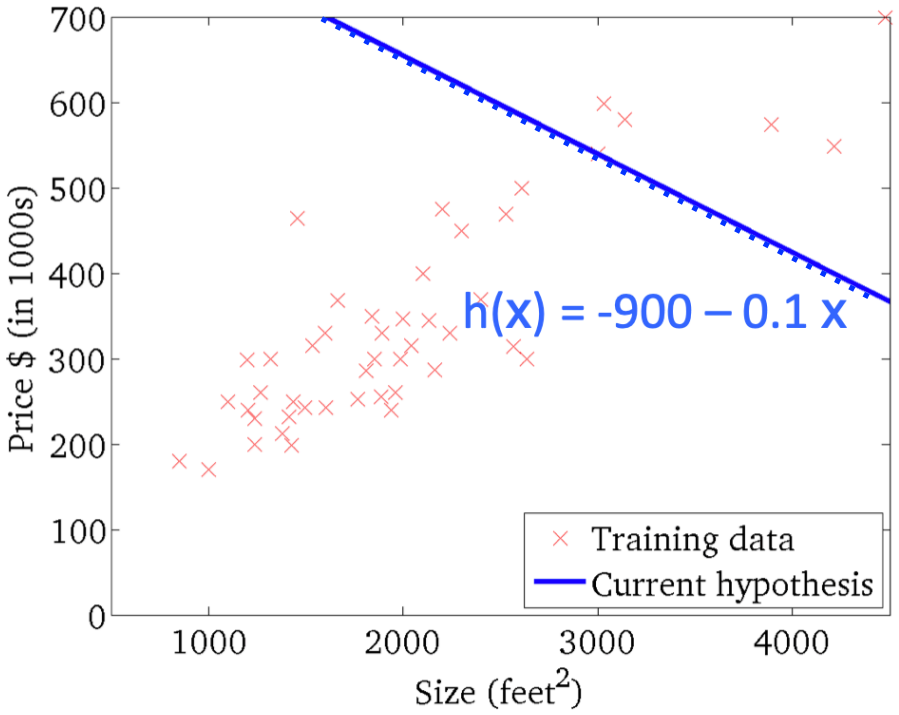
Hyperparameters:
- Initial point w_0
- Step size η



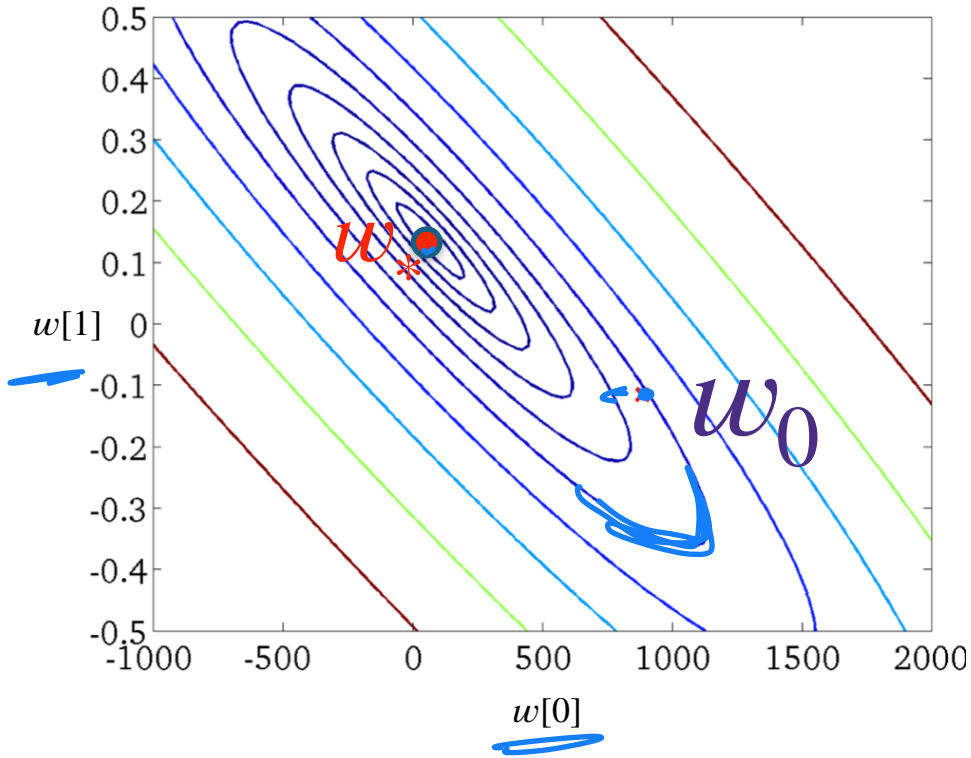
Note that as $t \to \infty$ we have $\frac{df(w)}{dw}\Big|_{w=w_t} \to 0$

1-dimensional linear regression with 2 parameters

$$\{(x_i, y_i)\}_{i=1}^{n}$$



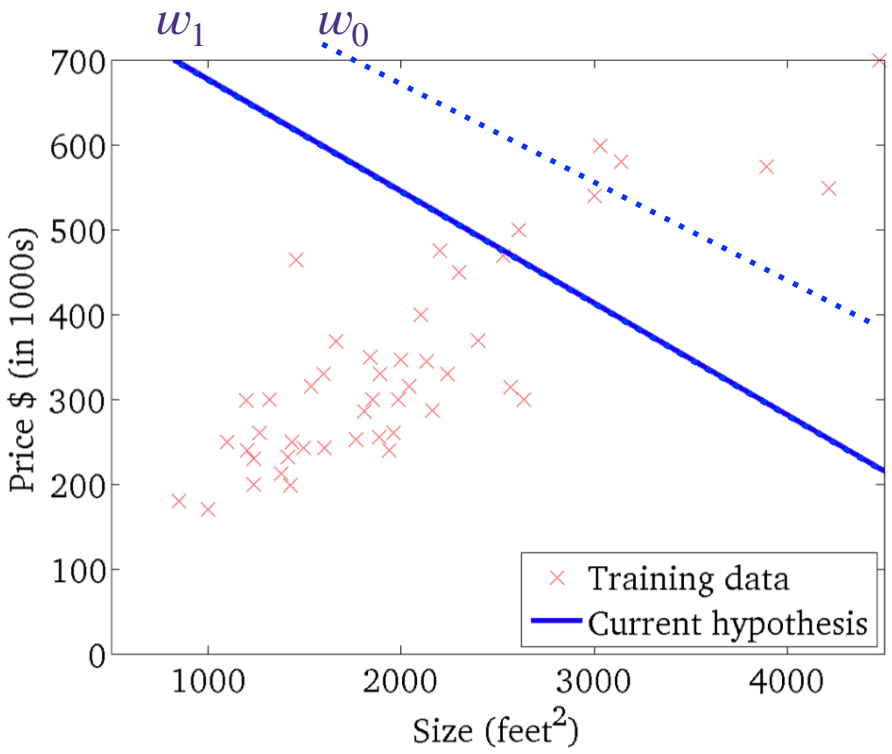
$$w_{t+1} \leftarrow w_t - \eta \cdot \nabla_w f(w_t)$$



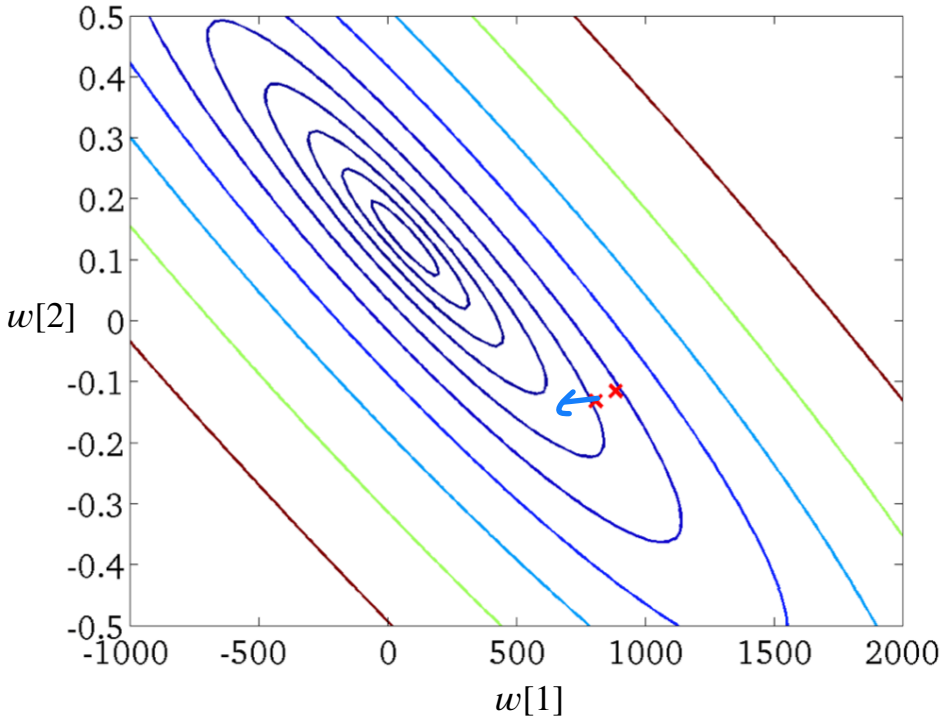
Evolution of the predictor y = w[0] + w[1]x

Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters

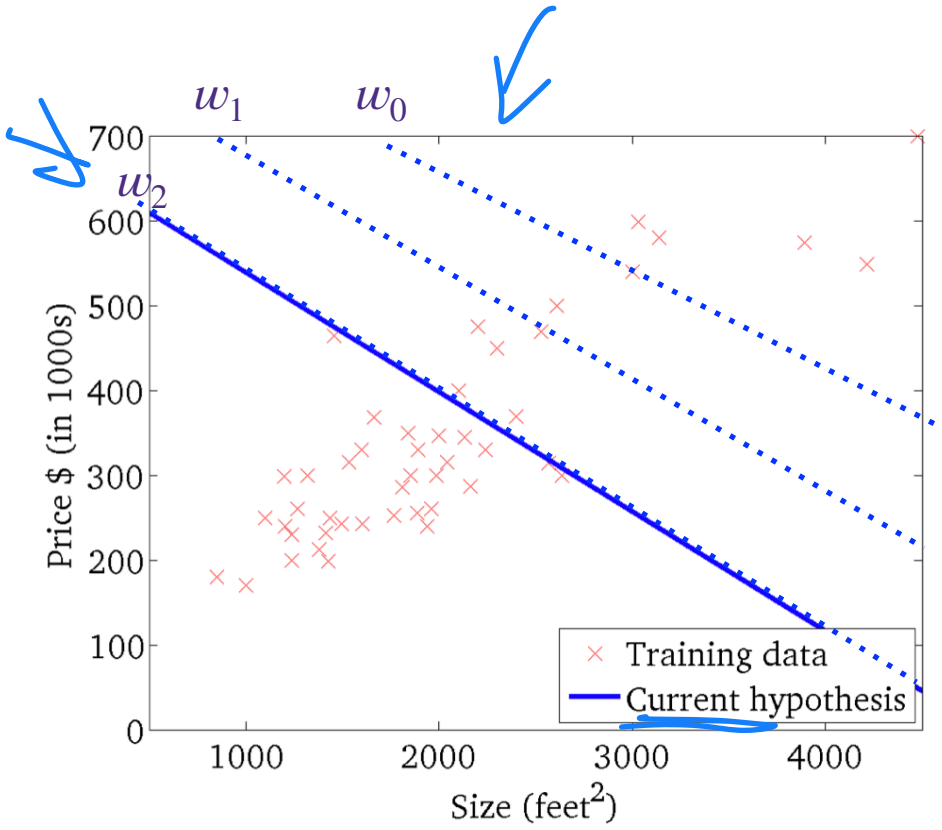


Evolution of the predictor y = w[0] + w[1]x

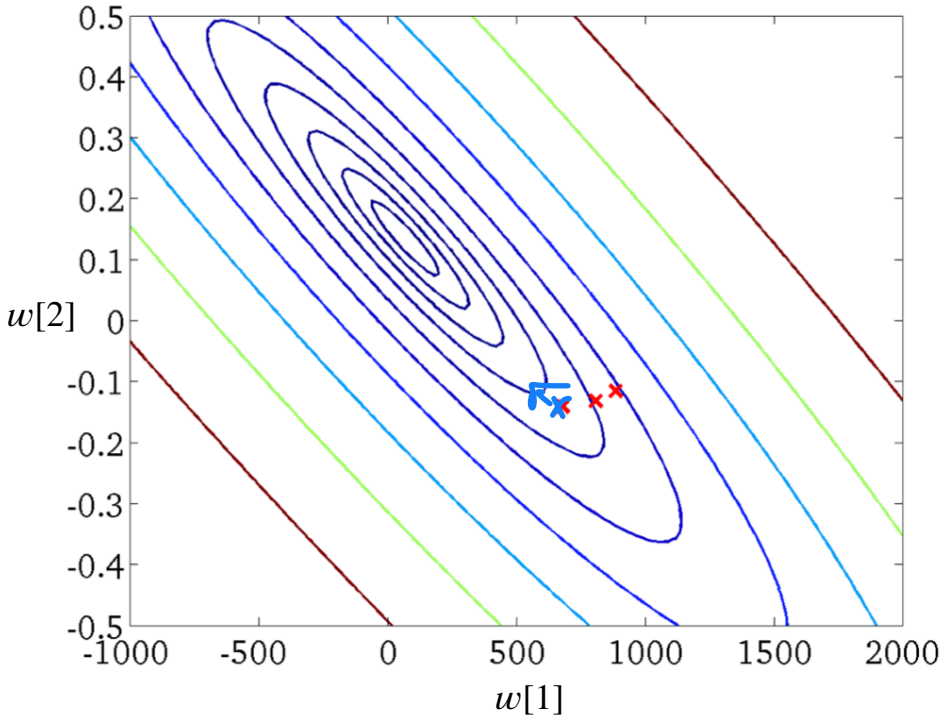


Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters

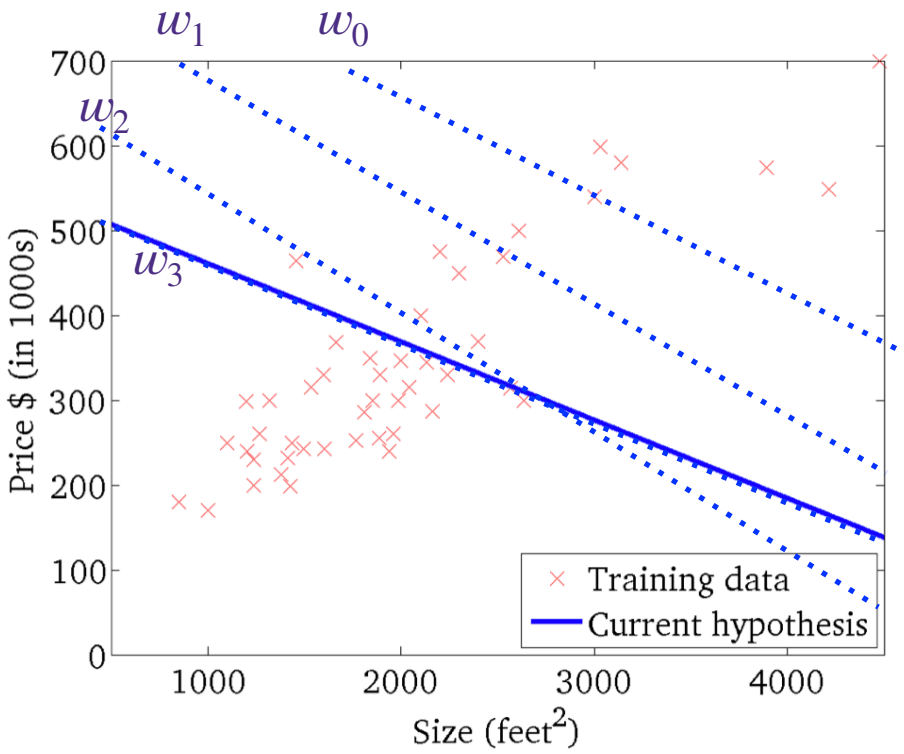


Evolution of the predictor y = w[0] + w[1]x

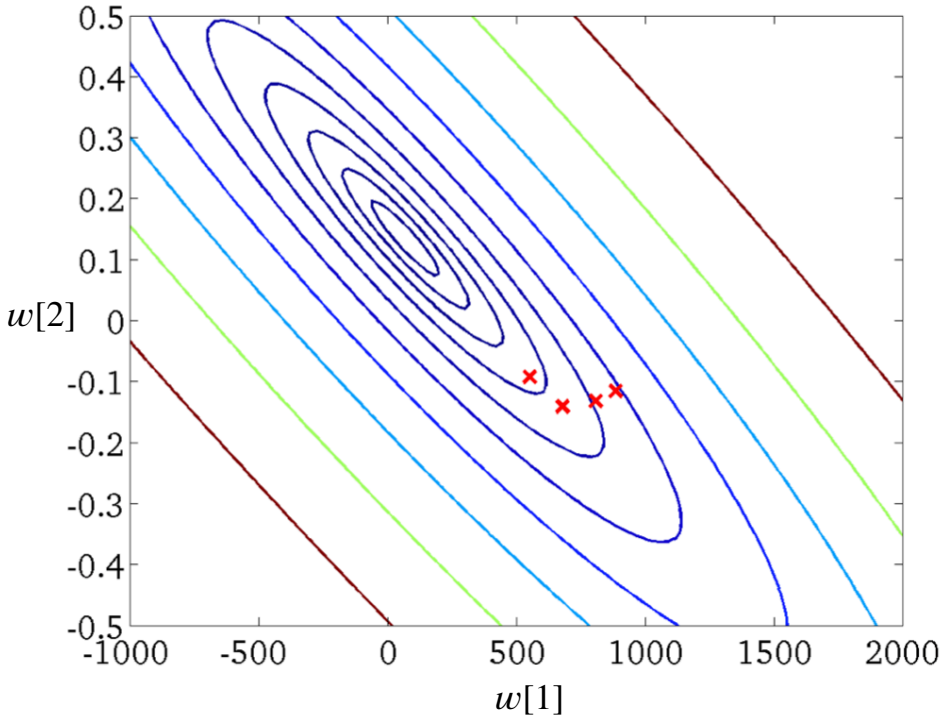


Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters

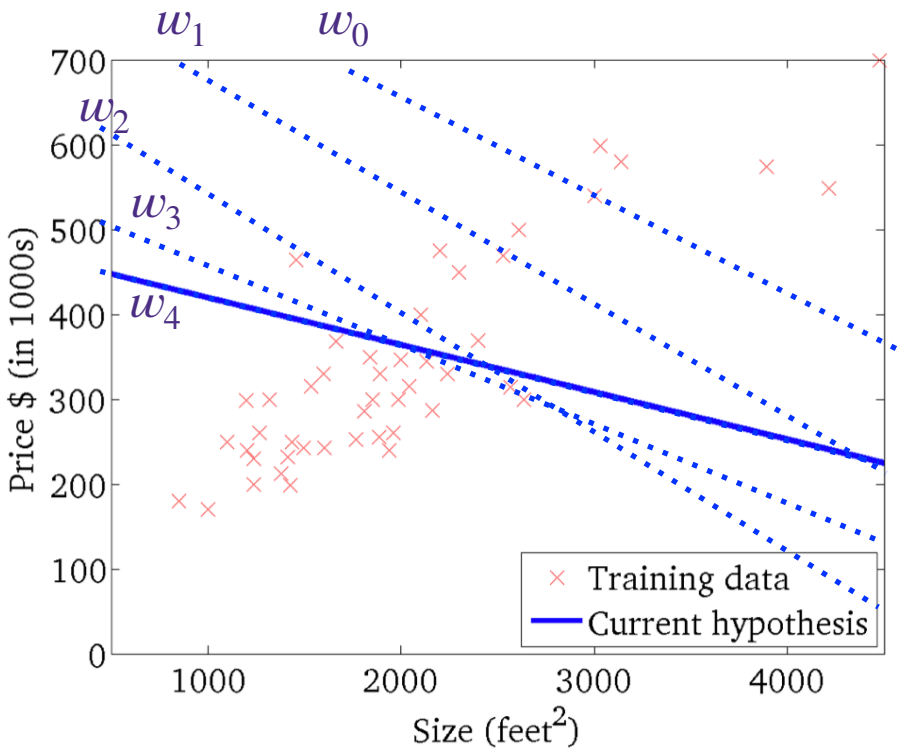


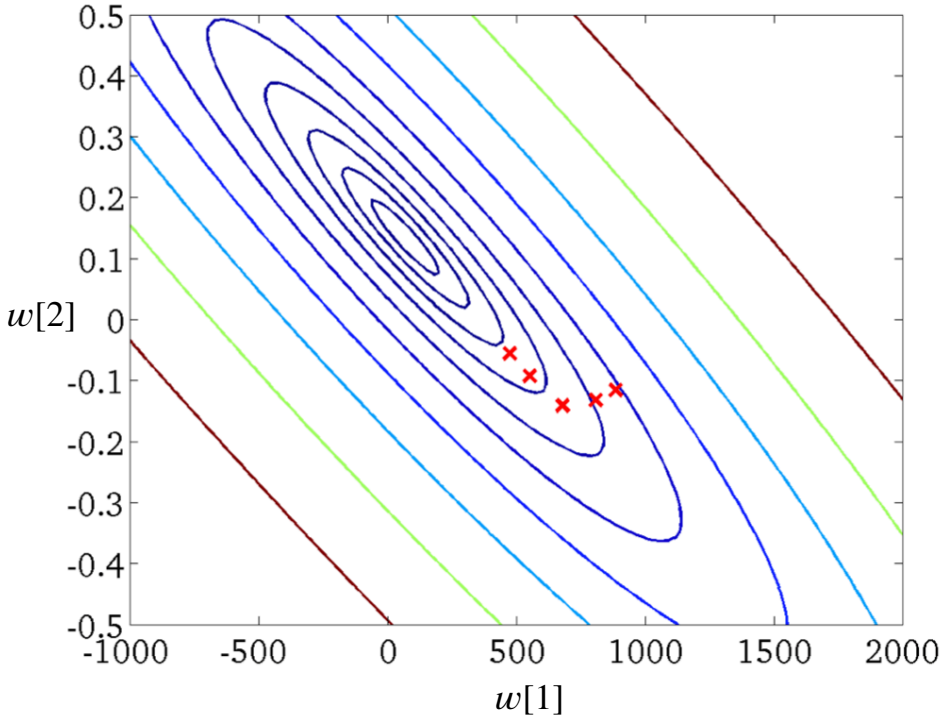
Evolution of the predictor y = w[0] + w[1]x



Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters

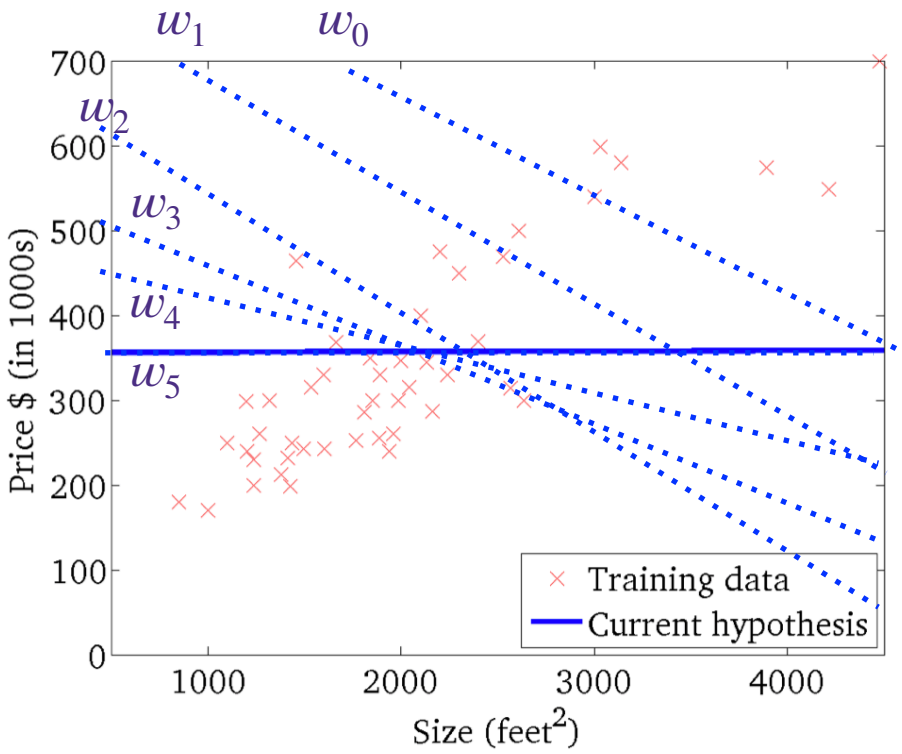




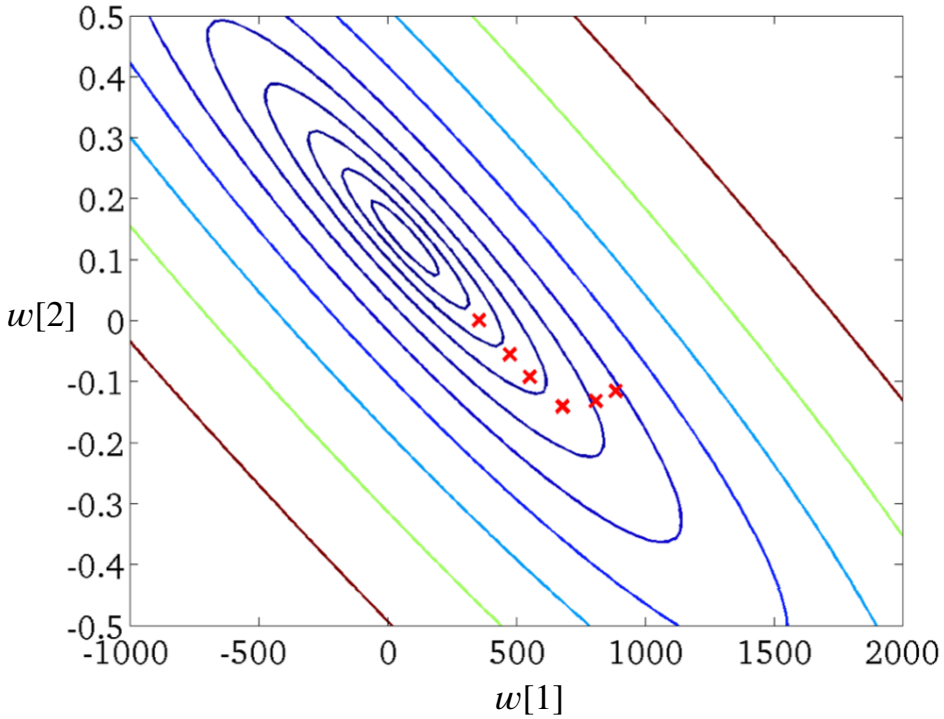
Evolution of the predictor y = w[0] + w[1]x

Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters

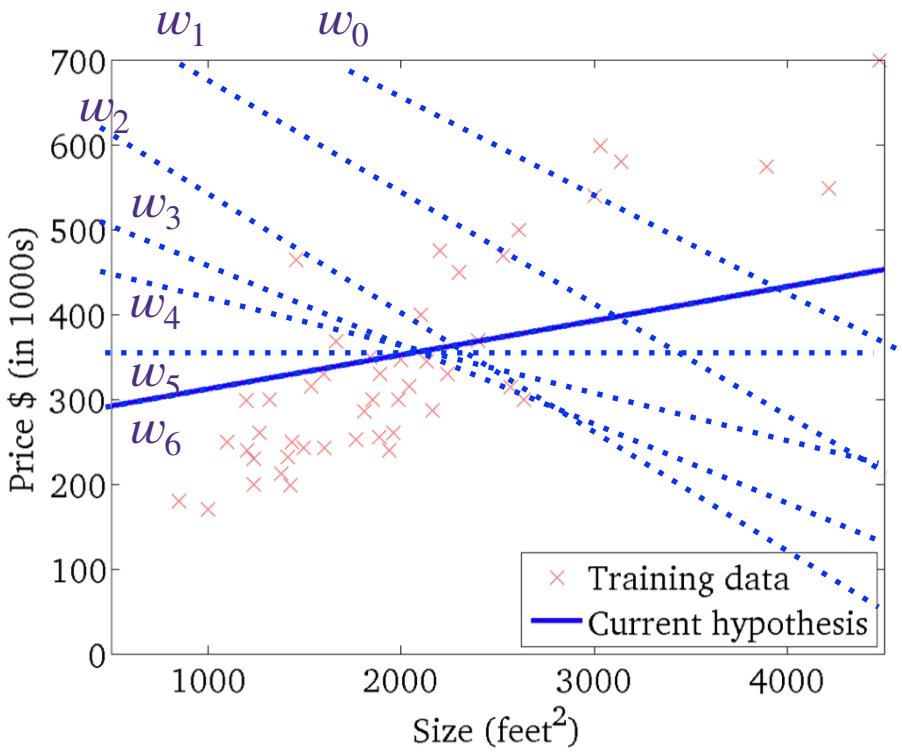


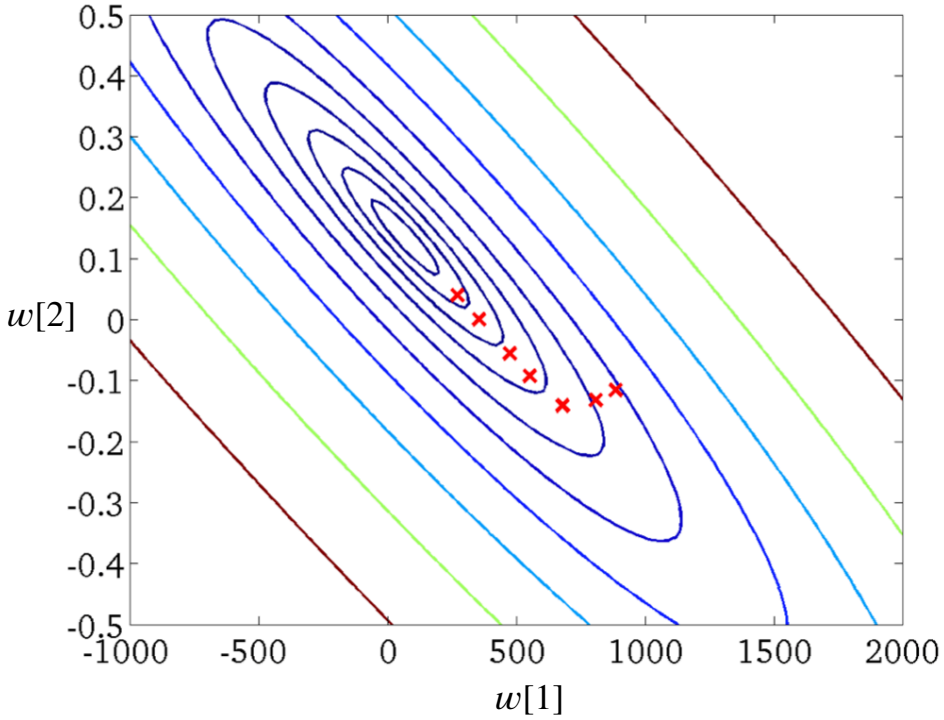
Evolution of the predictor y = w[0] + w[1]x



Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters

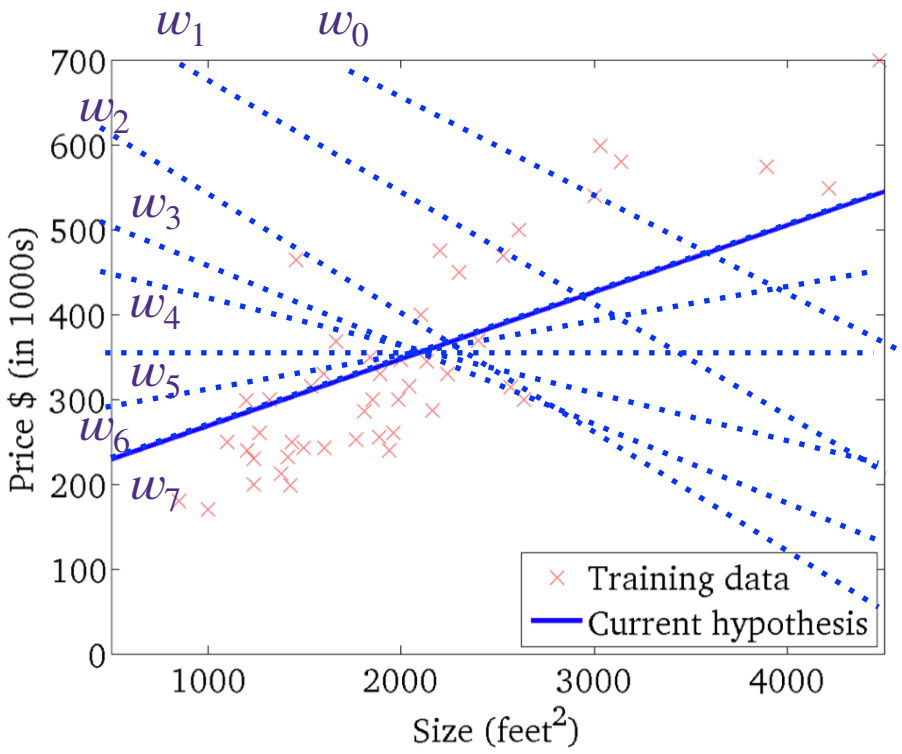




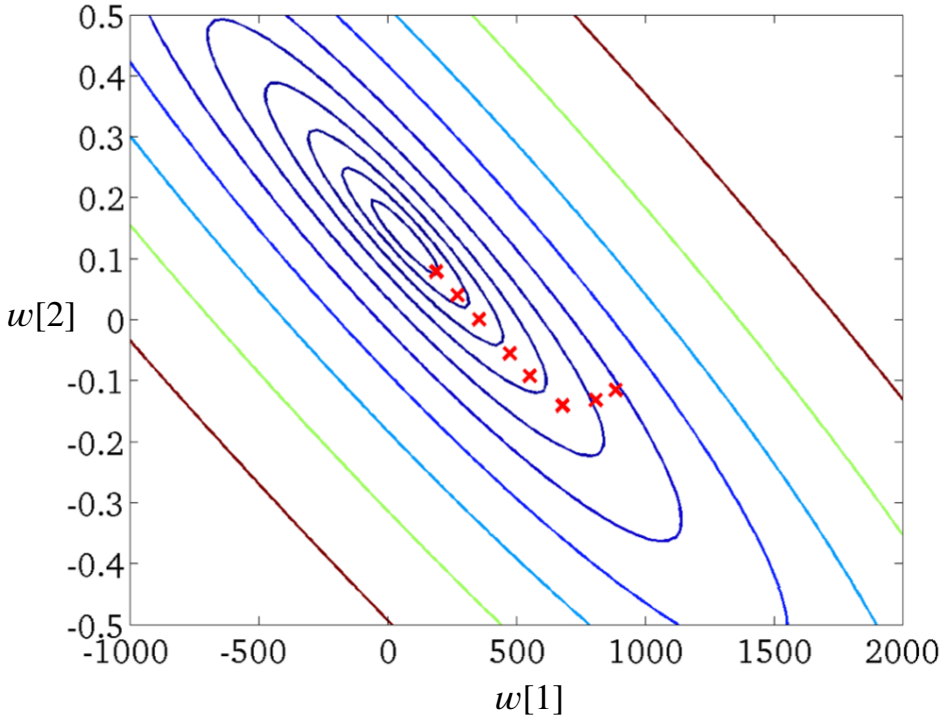
Evolution of the predictor y = w[0] + w[1]x

Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters

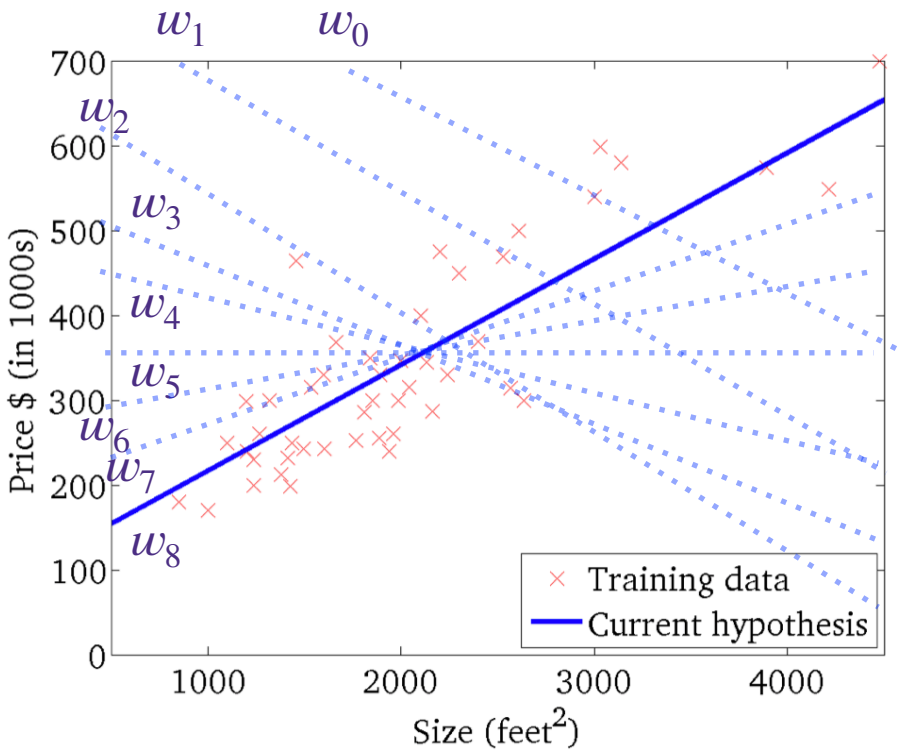


Evolution of the predictor y = w[0] + w[1]x

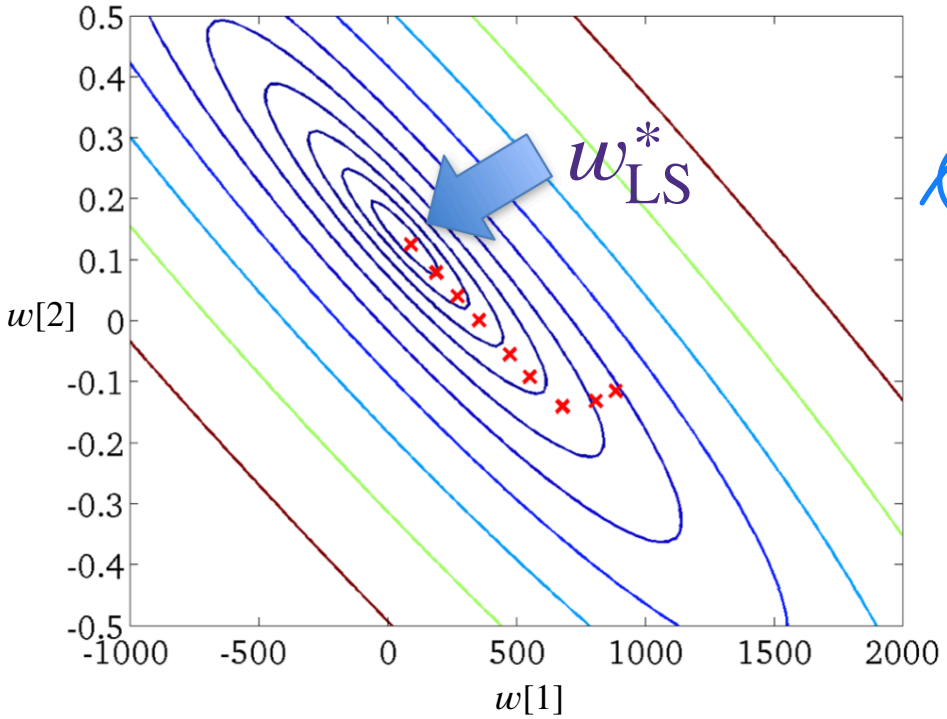


Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

1-dimensional linear regression with 2 parameters



Evolution of the predictor y = w[0] + w[1]x



Gradient descent dynamics in the parameter space w
Ovals show the **level set** of the objective function

Warmup: Quadratic functions

$$\hat{w} = \underset{w}{\text{argmin}} \; aw^2 + bw + c$$

→ f(w) (loss)

recall:

$$w_{t+1} = w_t - \eta \left. \frac{df(w)}{dw} \right|_{w=w_t}$$

$w_0 \sim N(0, I\sigma^2)$

$$\left. \frac{df(w)}{dw} \right|_{w=w_0} = 2aw_0 + b$$

$$w_1 = w_0 - \eta(2aw_0 + b)$$

Example: Linear regression

$f(w)$

$$\hat{w} = \underset{w}{\operatorname{argmin}} \frac{1}{2} \sum_i \left(y_i - x_i^\top w\right)^2$$

$$\nabla_w f(w_0) = X^\top (X w - y)$$

$d \times n \quad (n \times d \quad d \times 1 \quad - \quad n \times 1)$

$$w_{t+1} \leftarrow w_t - \eta X^\top (X w_0 - y)$$

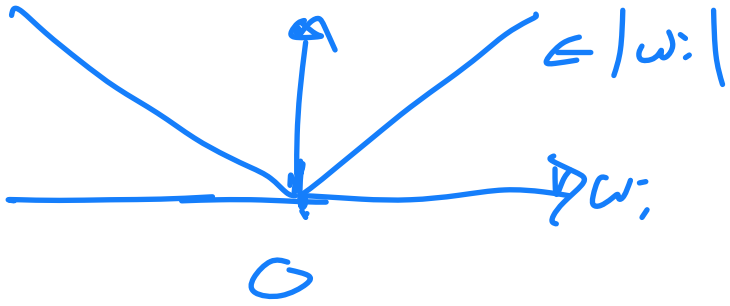
Example: Lasso

$$\hat{w} = \underset{w}{\operatorname{argmin}} \frac{1}{2}\|y - Xw\|_2^2 + \lambda\|w\|_1$$

? Double check

$$\nabla_w f = X^T(Xw - y) + \lambda \sum_{i=1}^{n} \operatorname{sign}(w_i)$$

$$w_{t+1} \leftarrow w_t - \eta \cdot \nabla_w f|_{w=w_t}$$



convex → local minima = global minimum

$$\frac{\partial |w_i|}{\partial w_i} = \begin{cases} +1 & w_i > 0 \\ [-1, 1] & w_i = 0 \\ -1 & w_i < 0 \end{cases}$$

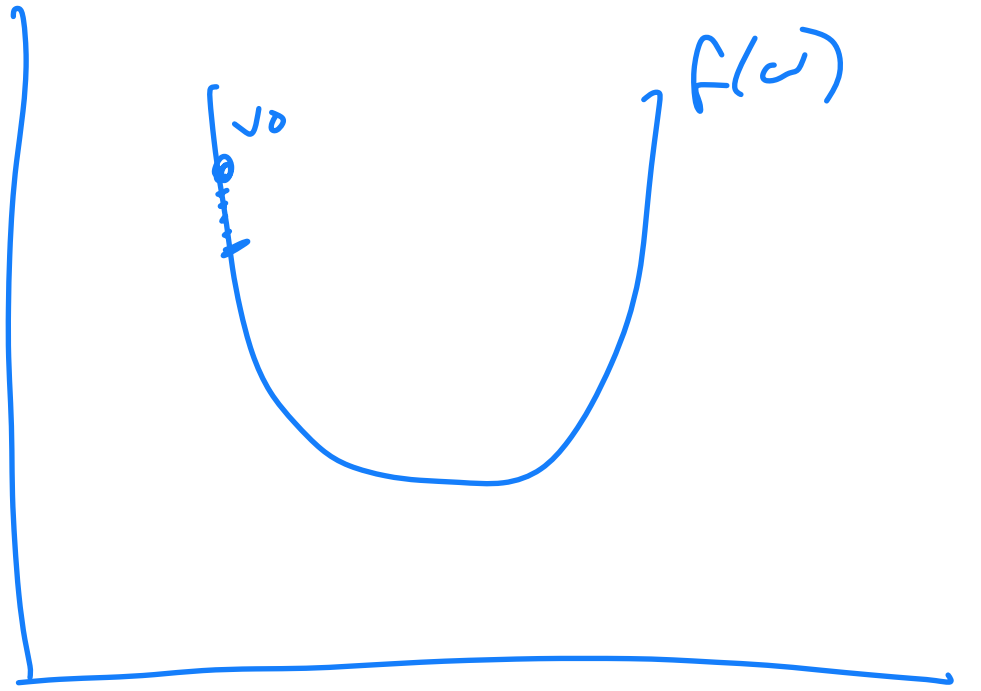
undefined → so define subgradient

$$f(w) = a w^2 + b w + c + |w|$$

$$\frac{df}{dw} = 2aw + b + \text{sign}(w) = 0 \qquad \text{// can find if } w \text{ is 1-d}$$

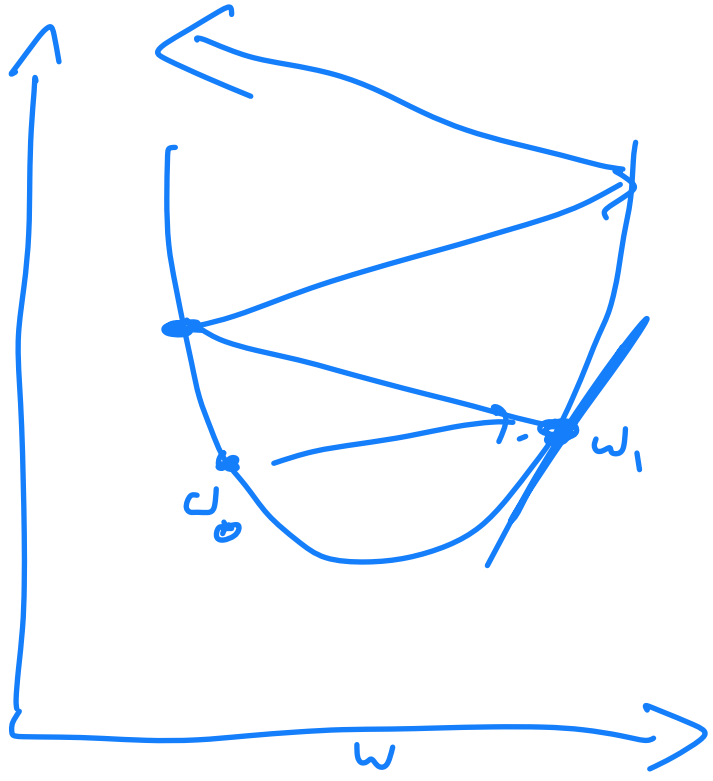
$w \in \mathbb{R}^d$ $\quad$ $\text{sign}(w) \rightarrow 2^d$ possibilities

How do you choose a step size?



step size η is too small

slow convergence

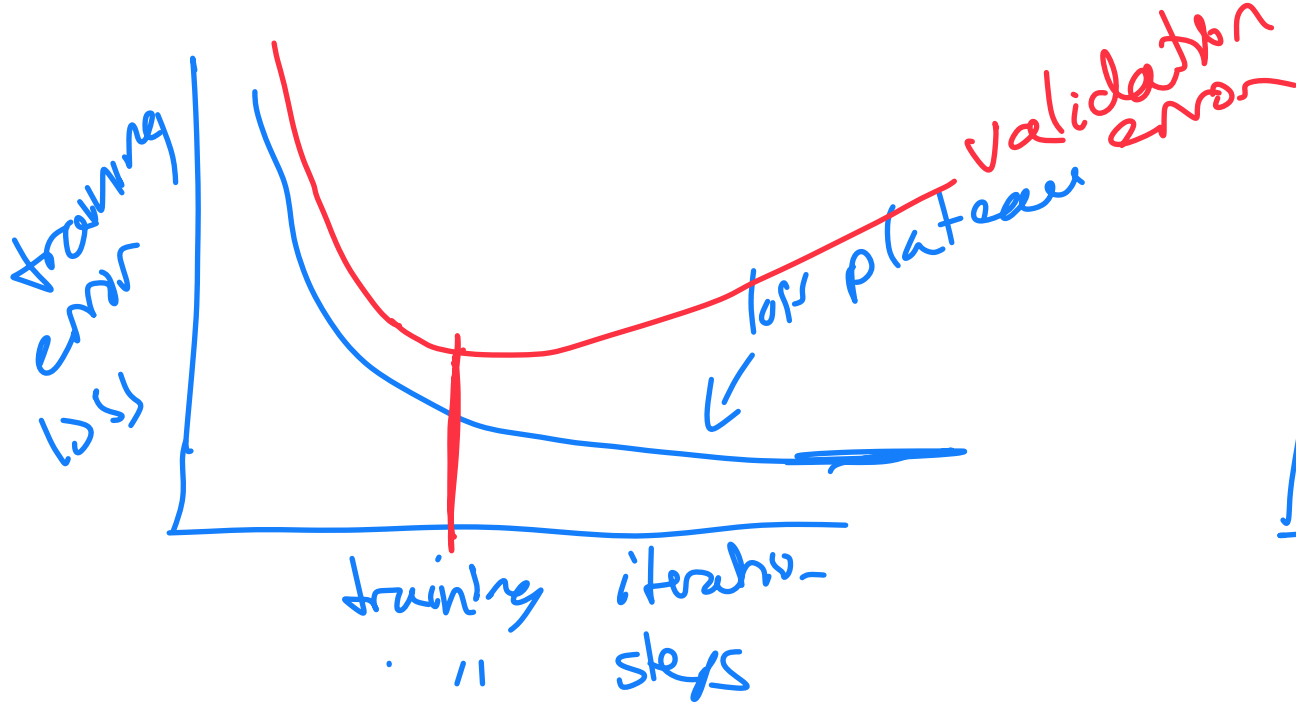


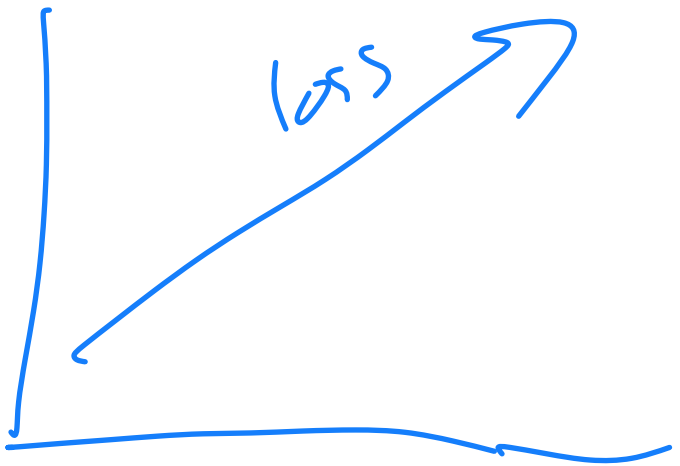
η is too big

- If η too small, converges very, very slowly.
- If η too big, does not converge!
- In practice: guess and check

How do you choose a step size?

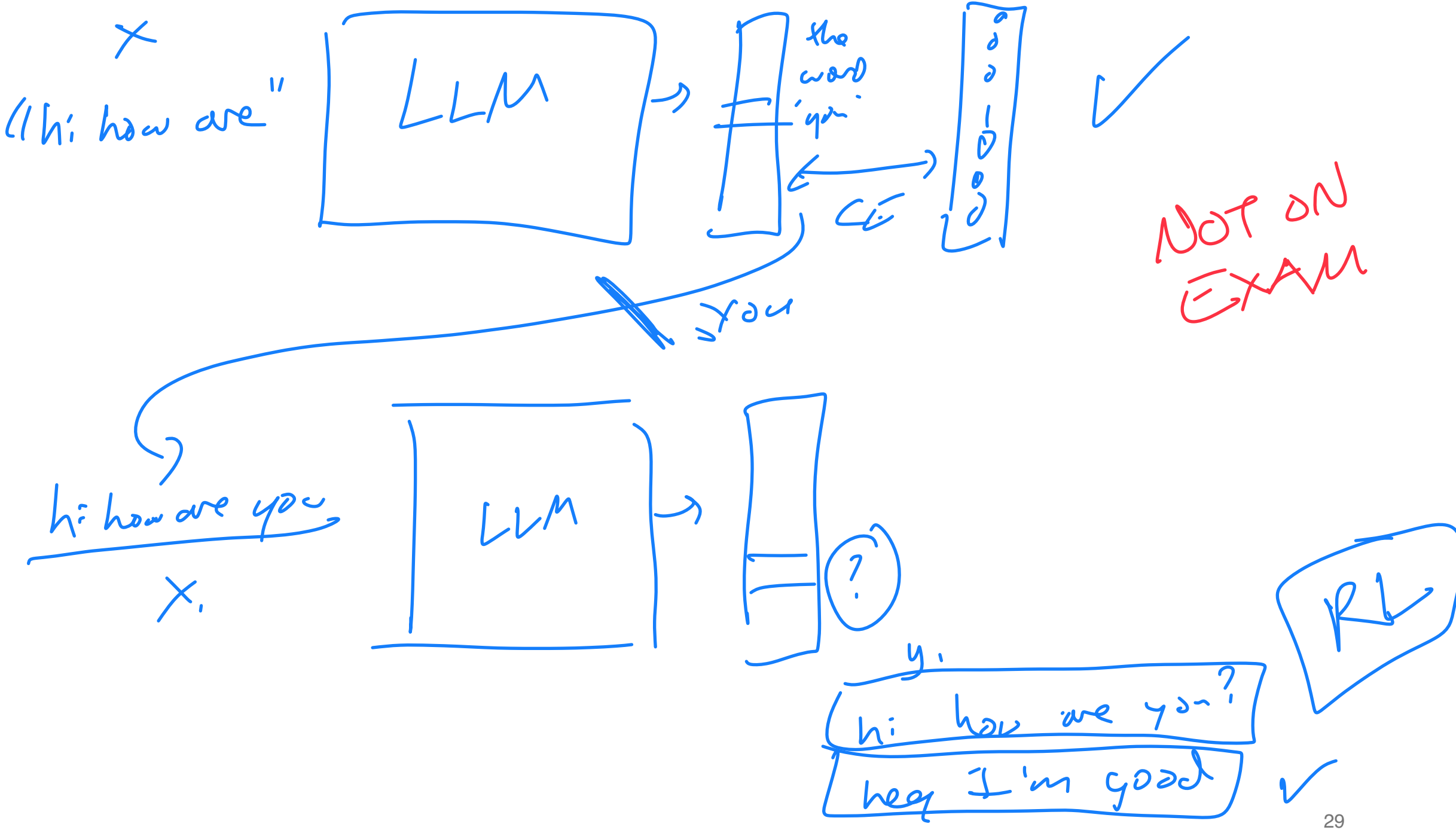
→ visualize loss over training!!





train loss effectively minimized, but use early stopping to prevent overfitting

loss increasing is a sign step size might be too big.


x
"hi how are"
LLM
the word 'you'
CE
you
NOT ON EXAM
hi how are you
x.
LLM
?
y.
hi how are you?
hey I'm good
RL