Lecture 14:
Stochastic Gradient Descent

-What do we use in practice?

Machine Learning Problems

- Given data: $\{(x_i, y_i)\}_{i=1}^n \qquad x_i \in \mathbb{R}^d \qquad y_i \in \mathbb{R}$
- Learning a model's parameters: $\frac{1}{n}\sum_{i=1}^n \ell_i(w) = \frac{1}{n}\sum_{i=1}^n \ell(f_w(x_i), y_i)$

- Gradient Descent (GD):
  one update takes cdn operations/time for some constant c>0

$$w_{t+1} \leftarrow w_t - \eta \frac{1}{n}\sum_{i=1}^n \nabla \ell_i(w_t)$$

- Stochastic Gradient Descent (SGD): one update takes cd operations/time

$$w_{t+1} \leftarrow w_t - \eta \nabla \ell_{I_t}(w_t)$$

I_t drawn uniform at random from $\{1, \ldots, n\}$

- SGD is an unbiased estimate of the GD

$$\mathbb{E}[\nabla \ell_{I_t}(w)] = \sum_{i=1}^n \mathbb{P}(I_t = i)\nabla \ell_i(w) = \frac{1}{n}\sum_{i=1}^n \nabla \ell_i(w)$$

Stochastic Gradient Descent

Theorem

Let $w_{t+1} = w_t - \eta \nabla_w \ell_{I_t}(w)\Big|_{w=w_t}$ I_t drawn uniform at random from $\{1, \dots, n\}$ so that

$$\mathbb{E}\left[\nabla \ell_{I_t}(w)\right] = \frac{1}{n}\sum_{i=1}^{n} \nabla \ell_i(w) =: \nabla \ell(w)$$

If $\|w_0 - w_*\|_2^2 \le R$ and $\sup_w \max_i \|\nabla \ell_i(w)\|_2 \le G$ then

after T steps of SGD with stepsize η, we achieve $\mathbb{E}[\ell(\overline{w}) - \ell(w_*)] \le \frac{R}{2T\eta} + \frac{\eta G}{2}$

Selecting the optimal stepsize, $\min_{\eta>0} \frac{R}{2T\eta} + \frac{\eta G}{2} = \sqrt{\frac{RG}{T}}$ for $\eta = \sqrt{\frac{R}{GT}}$

$$\overline{w} = \frac{1}{T}\sum_{t=1}^{T} w_t$$

(In practice use last iterate)

Convergence rate: $O\left(\frac{1}{\sqrt{T}}\right)$

(Fixed optimal step size)

$$-\frac{R}{2T\eta^2} + \frac{G}{2} = 0$$

Taking the derivative of RHS to zero

We want to show that

$$\mathbb{E}\left[\ell\left(\frac{1}{T}\sum_{t=1}^{T} w_t\right) - \ell(w_*)\right] \le \mathbb{E}\left[\frac{1}{T}\sum_{i=1}^{T} \ell(w_t) - \ell(w_*)\right]$$

← Follows from convexity of $\ell(\cdot)$ and Jensen's inequality (3 slides later)

$$\le \frac{1}{T}\sum_{i=1}^{T} \mathbb{E}\left[\ell(w_t) - \ell(w_*)\right]$$

← Follows from linearity of expectation

$$\le \frac{R}{2T\eta} + \frac{\eta G}{2}$$

← We are left to show this

Proof $\mathbb{E}[||w_{t+1} - w_*||_2^2] = \mathbb{E}[||w_t - \eta \nabla \ell_{I_t}(w_t) - w_*||_2^2]$

$= \mathbb{E}[\|w_t - w_*\|^2] + \eta^2 \mathbb{E}[\|\nabla \ell_{I_t}(w_t)\|^2] - 2\eta \mathbb{E}[\nabla \ell_{I_t}(w_t)](w_t - w_*)$

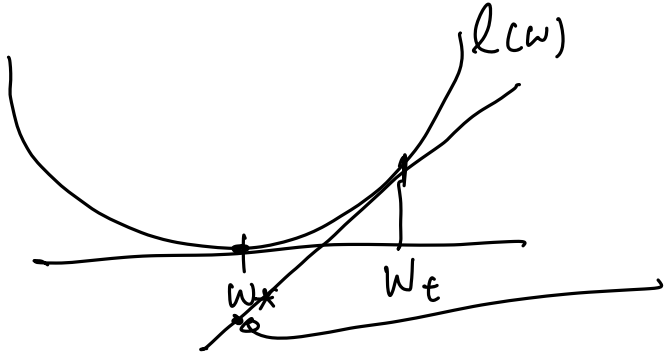
($\le G$)

Previous Error Grad noise error

$-2\eta\,[\ell(w_t)^T(w_t - w_*)]$

$\le -2\eta \cdot (\ell(w_t) - \ell(w_*))$

Gain due to Grad step.



$\ell(w_t) + \nabla\ell(w_t)^T \cdot (w_* - w_t) \le \ell(w_*)$

Stochastic Gradient Descent

Proof

$$\mathbb{E}[||w_{t+1} - w_*||_2^2] = \mathbb{E}[||w_t - \eta \nabla \ell_{I_t}(w_t) - w_*||_2^2]$$
$$\leq \mathbb{E}[\|w_t - w_*\|_2^2] + \eta^2 G - 2\eta(\ell(w_t) - \ell(w_*))$$

$$\ell(w_t) - \ell(w_*) \leq \left(\underbrace{\mathbb{E}[\|w_t - w_*\|_2^2] - \mathbb{E}[\|w_{t+1} - w_*\|_2^2]}_{\text{Telescoping.}} + \eta^2 G \right) \frac{1}{2\eta}$$

$$\sum_{t=1}^{T} (\ell(w_t) - \ell(w_*)) \leq \left(\mathbb{E}[\|w_0 - w_*\|_2^2] - \mathbb{E}[\|w_T - w_*\|_2^2] + T \cdot \eta^2 G \right) \frac{1}{2\eta}$$
$$\leq \frac{R}{2\eta} + \frac{T\eta G}{2}$$

Stochastic Gradient Descent

Proof

$$\mathbb{E}[||w_{t+1} - w_*||_2^2] = \mathbb{E}[||w_t - \eta \nabla \ell_{I_t}(w_t) - w_*||_2^2]$$

$$= \mathbb{E}[||w_t - w_*||_2^2] - 2\eta \mathbb{E}[\nabla \ell_{I_t}(w_t)^T (w_t - w_*)] + \eta^2 \mathbb{E}[||\nabla \ell_{I_t}(w_t)||_2^2]$$

$$\leq \mathbb{E}[||w_t - w_*||_2^2] - 2\eta \mathbb{E}[\ell(w_t) - \ell(w_*)] + \eta^2 G$$

$$\mathbb{E}[\nabla \ell_{I_t}(w_t)^T (w_t - w_*)] = \mathbb{E}\big[\mathbb{E}[\nabla \ell_{I_t}(w_t)^T (w_t - w_*) | I_1, w_1, \ldots, I_{t-1}, w_{t-1}]\big]$$

$$= \mathbb{E}\big[\nabla \ell(w_t)^T (w_t - w_*)\big]$$

$$\geq \mathbb{E}\big[\ell(w_t) - \ell(w_*)\big]$$

$$\sum_{t=1}^{T} \mathbb{E}[\ell(w_t) - \ell(w_*)] \leq \frac{1}{2\eta} \left(\mathbb{E}[||w_1 - w_*||_2^2] - \mathbb{E}[||w_{T+1} - w_*||_2^2] + T\eta^2 G\right)$$

$$\leq \frac{R}{2\eta} + \frac{T\eta G}{2}$$

We have shown: $\sum_{t=1}^{T} \mathbb{E}[\ell(w_t) - \ell(w_*)] \le \frac{R}{2\eta} + \frac{T\eta G}{2}$

Jensen's inequality

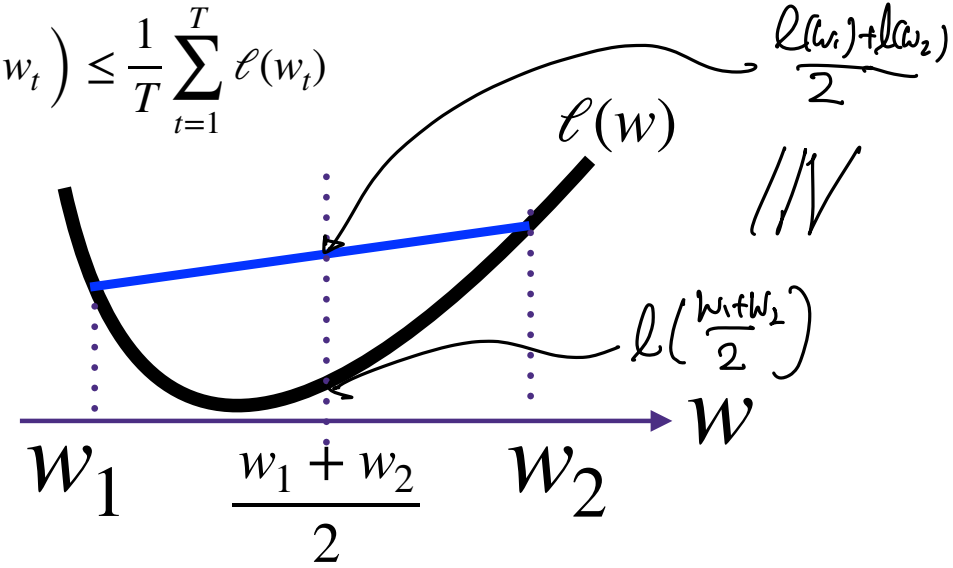
$$\mathbb{E}[\ell(\bar{w}) - \ell(w_*)] \le \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\ell(w_t) - \ell(w_*)] \qquad \bar{w} = \frac{1}{T}\sum_{t=1}^{T} w_t$$

$$\le \frac{R}{2\eta T} + \frac{\eta G}{2}$$

Jensen's inequality:

For any $\{w_1, \ldots, w_T\}$ and a convex function $\ell(\cdot)$, we have

$$\ell\left(\frac{1}{T}\sum_{t=1}^{T} w_t\right) \le \frac{1}{T}\sum_{t=1}^{T} \ell(w_t)$$



Mini-batch SGD

- Instead of one iterate, average B stochastic gradient together

- Advantages:
  - Smaller variance: the variance of the stochastic gradient is smaller by a factor of $1/\sqrt{B}$
  - Parallelization: each gradient in the mini-batch can be computed in parallel

- If you have regularizer, $\frac{1}{n}\sum_{i=1}^{n} \ell_i(w) + r(w)$, then update with the stochastic gradient of the loss and gradient of the regularizer

Questions?

Lecture 14: Coordinate Descent

- How to solve non-smooth optimization like Lasso?

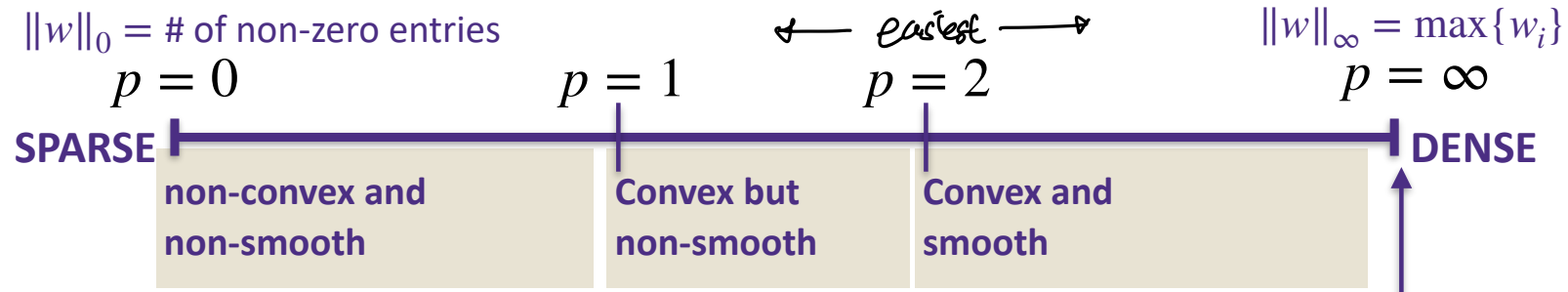
$$\hat{w}_{\text{Lasso}} = \arg\min_{w \in \mathbb{R}^d} \underbrace{\|\mathbf{y} - \mathbf{X}w\|_2^2 + \lambda \|w\|_1}_{f(w)}$$

Sparsity/Complexity tradeoff

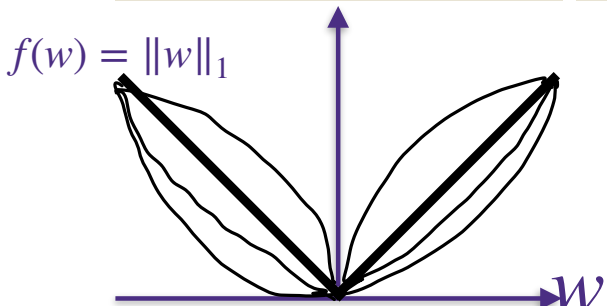
- ℓ_p-norm of a vector is defined as $\|w\|_p \triangleq \left(|w_1|^p + |w_2|^p + \cdots + |w_d|^p\right)^{1/p}$
- Consider regularized least squares problem of minimizing

$$\mathscr{L}(w) = \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \lambda \|w\|_p^p$$

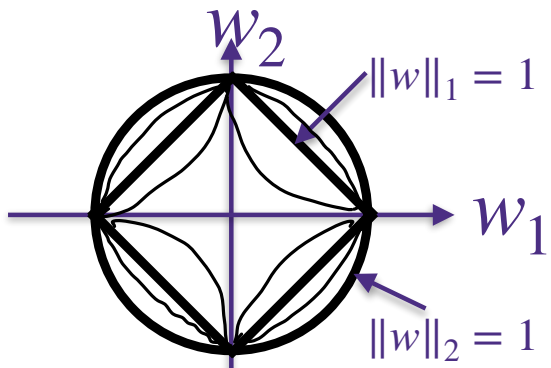
- This is ridge regression for $p = 2$ and Lasso for $p = 1$



convex but non-smooth



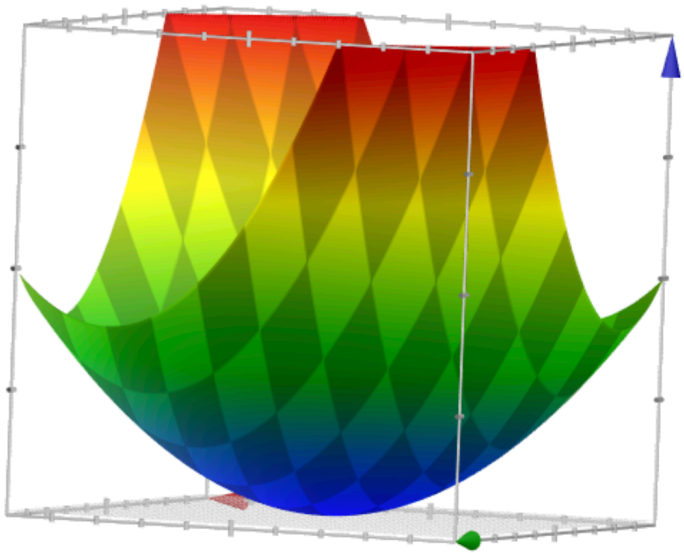
Non-convex and non-smooth functions are slower to optimize

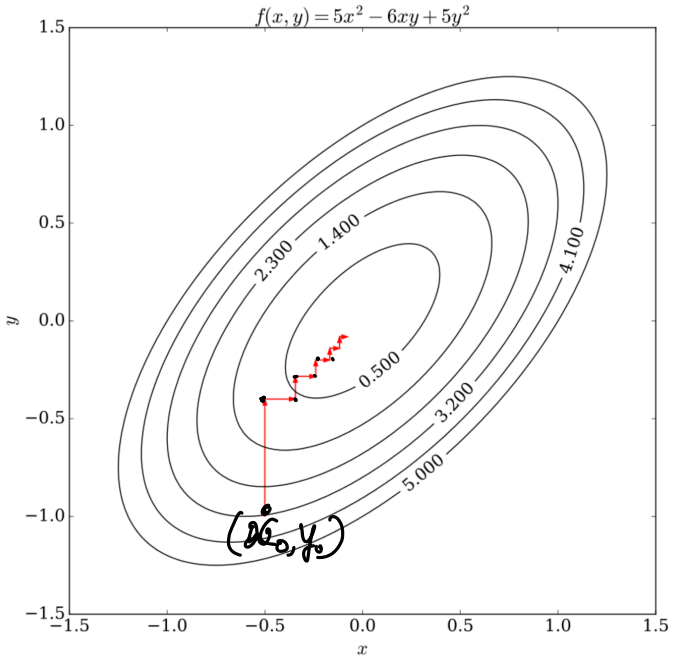


More pointy level set gives sparser solution

Optimization: how do we solve Lasso?

- among many methods to find the solution, we will learn **coordinate descent method**
- as an illustrating example, we show coordinate descent updates on finding the minimum of a very simple function:
  $f(x, y) = 5x^2 - 6xy + 5y^2$





How do we solve Lasso: $\min_w \mathscr{L}(w) + \lambda \|w\|_1$?

- **Coordinate descent**
  - **input: training data S_{train}, max # of iterations T**
  - **initialize: $w^{(0)} = \mathbf{0} \in \mathbb{R}^d$**
  - **for $t = 1, \ldots, T$**
    - **for $j = 1, \ldots, d$**
      - **fix $w_1^{(t)}, \ldots, w_{j-1}^{(t)}$ and $w_{j+1}^{(t-1)}, \ldots, w_d^{(t-1)}$, and**

$$w_j^{(t)} \leftarrow \arg\min_{w_j \in \mathbb{R}} \mathscr{L}\left(\begin{bmatrix} w_1^{(t)} \\ \vdots \\ w_{j-1}^{(t)} \\ w_j \\ w_{j+1}^{(t-1)} \\ \vdots \\ w_d^{(t-1)} \end{bmatrix}\right) + \lambda \left\| \begin{bmatrix} w_1^{(t)} \\ \vdots \\ w_{j-1}^{(t)} \\ w_j \\ w_{j+1}^{(t-1)} \\ \vdots \\ w_d^{(t-1)} \end{bmatrix} \right\|_1$$

- **This inner step is a one-dimensional optimization, which is much easier to solve**

Coordinate descent for (un-regularized) linear regression

- let us understand what coordinate descent does on a simpler problem of linear least squares, which minimizes

$$\text{minimize}_w \mathscr{L}(w) = \|\mathbf{X}w - \mathbf{y}\|_2^2$$

- note that we know that the optimal solution is

$$\hat{w}_{\text{LS}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^\mathbf{T}\mathbf{y}$$

so we do not need to run any optimization algorithm

- we are solving this problem with **coordinate descent** as a starting example to learn how coordinate descent works

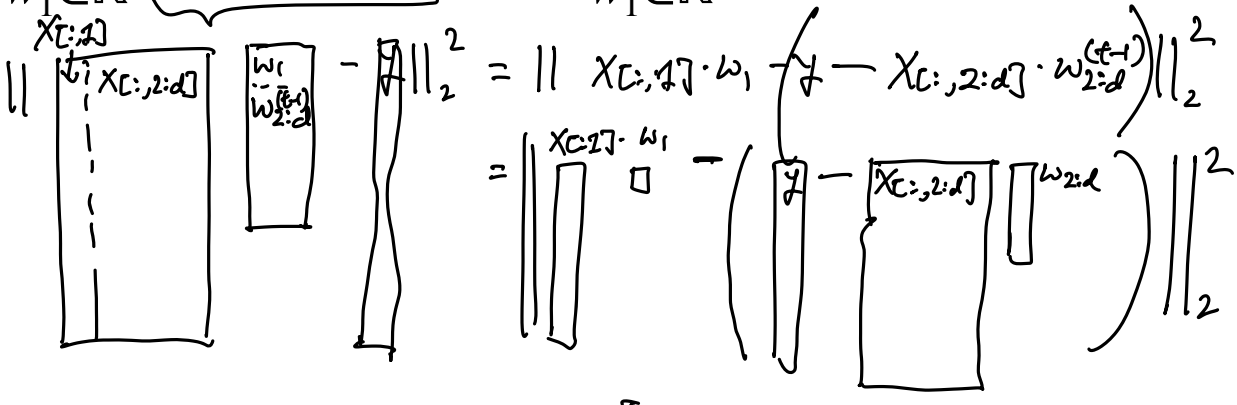
- the main challenge we address is, how do we update $w_j^{(t)} \in \mathbb{R}$?

- let us derive an **analytical rule** (i.e., closed form solution) for updating $w_j^{(t)}$, which generalizes to the case when we have Lasso regularizer.

Coordinate descent for (un-regularized) linear regression

We consider the case when updating coordinate $j = 1$

$$\min_{w_1 \in \mathbb{R}} \|\mathbf{X}w - \mathbf{y}\|_2^2 = \min_{w_1 \in \mathbb{R}} (aw_1 - b)^2 + \text{constant}$$



$$= x_{[:,1]}^T x_{[:,1]} \cdot w_1^2 - 2 \cdot w_1 \cdot x_{[:,1]}^T (y - X_{[:,2:d]} \cdot w_{2:d}^{(t-1)}) + \text{constant}$$

$$a = \sqrt{x_1^T x_1}, \quad b = \frac{x_1^T (y - X_{-1} \cdot w_{-1}^{(t-1)})}{\sqrt{x_1^T x_1}}$$

$$w_1^{(t)} = \frac{b}{a} = \frac{x_1^T (y - X_{-1} w_{-1}^{(t-1)})}{x_1^T x_1}$$

Coordinate descent for (un-regularized) linear regression

- we will study the case $j = 1$, for now (other cases are almost identical)
- when updating $w_1^{(t)}$, recall that

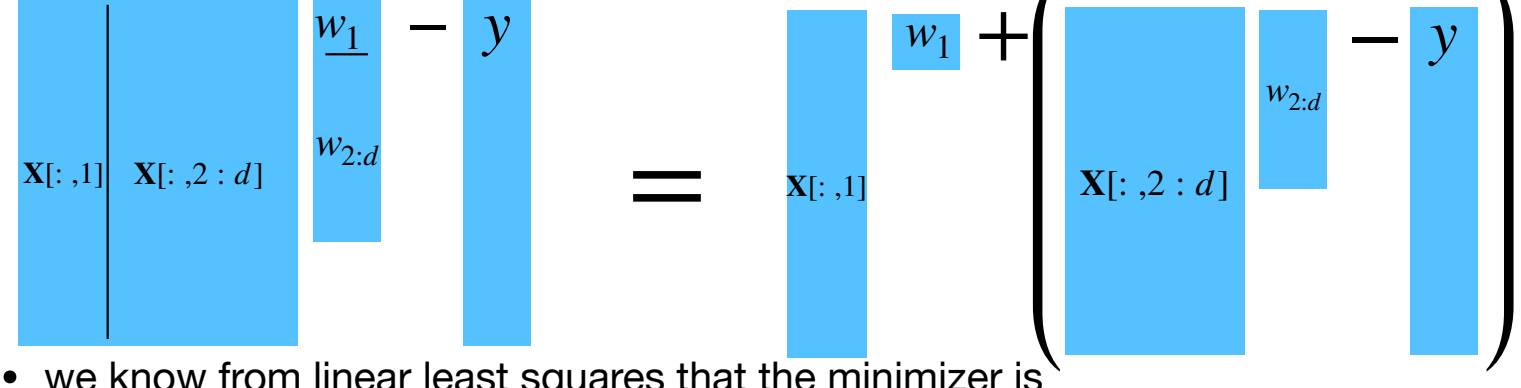
$$w_1^{(t)} \leftarrow \arg\min_{w_1} \|\mathbf{X}w - \mathbf{y}\|_2^2$$

where $w = [w_1, \; w_2^{(t-1)}, \; \ldots \; , w_d^{(t-1)}]^T$

- first step is to write the objective function in terms of the variable we are optimizing over, that is w_1:

$$\mathscr{L}(w) = \left\| \mathbf{X}[:,1]w_1 + \mathbf{X}[:,2:d]w_{2:d} - \mathbf{y} \right\|_2^2$$

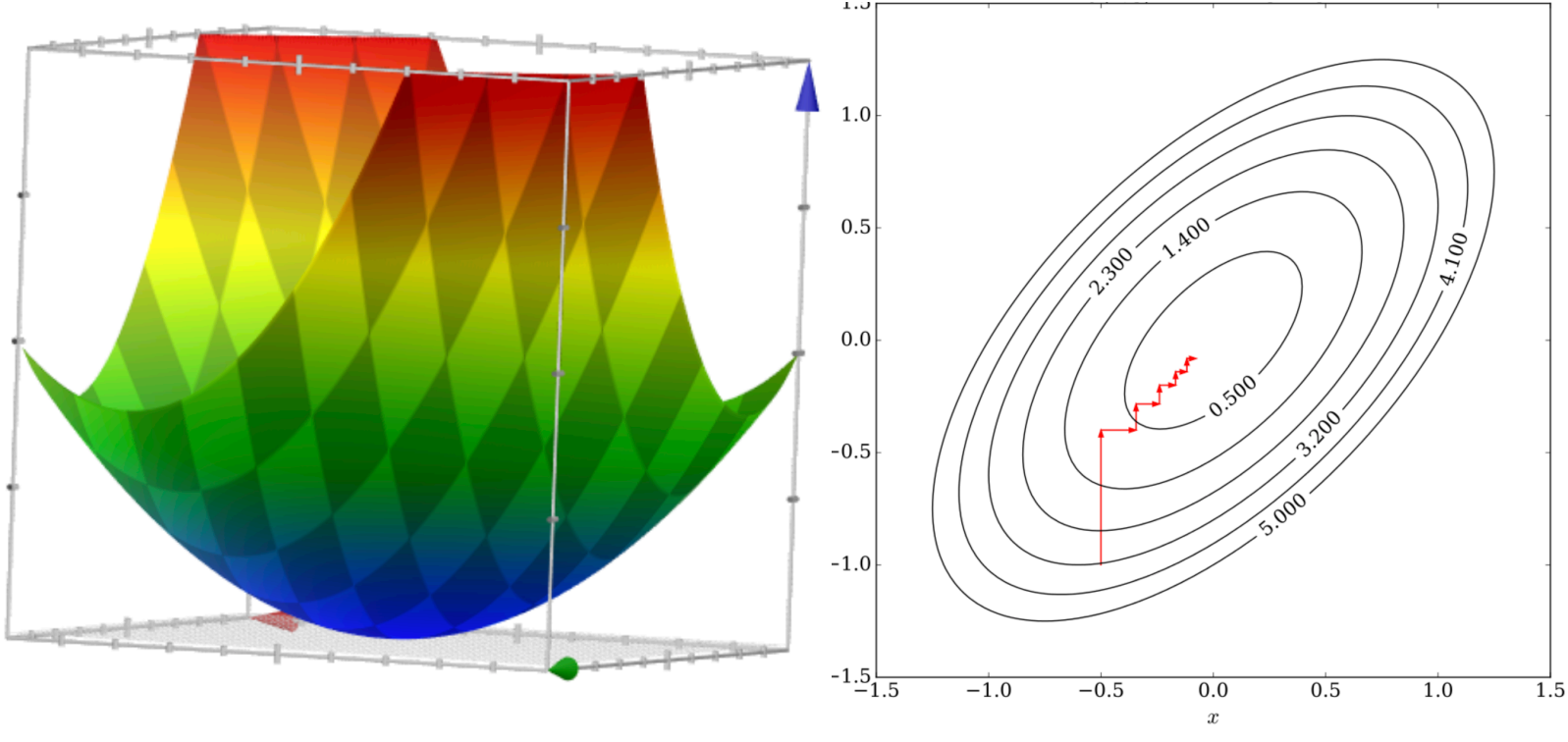
where $w_{2:d} = [w_2^{(t-1)}, \; \ldots \; , w_d^{(t-1)}]^T$



- we know from linear least squares that the minimizer is

$$w_1^{(t)} \leftarrow (\mathbf{X}[:,1]^T\mathbf{X}[:,1])^{-1}\,\mathbf{X}[:,1]^T\left(\mathbf{y} - \mathbf{X}[:,2:d]w_{2:d}\right)$$

- Coordinate descent applied to a quadratic loss



Coordinate descent for Lasso

- let us apply coordinate descent on Lasso, which minimizes

$$\text{minimize}_w \mathscr{L}(w) + \lambda \|w\|_1 = \|\mathbf{X}w - \mathbf{y}\|_2^2 + \lambda \|w\|_1$$

- the goal is to derive an **analytical rule** for updating $w_j^{(t)}$'s

- let us first write the update rule explicitly for $w_1^{(t)}$
  - first step is to write the loss in terms of w_1

$$\left\|\mathbf{X}[:,1]w_1 - \left(\mathbf{y} - \mathbf{X}[:,2:d]w_{2:d}\right)\right\|_2^2 + \lambda\Big(|w_1| + \underbrace{\|w_{2:d}\|_1}_{\text{constant}}\Big)$$

  - hence, the coordinate descent update boils down to

$$w_1^{(t)} \leftarrow \arg\min_{w_1} \underbrace{\left\|\mathbf{X}[:,1]w_1 - \left(\mathbf{y} - \mathbf{X}[:,2:d]w_{2:d}^{(t-1)}\right)\right\|_2^2 + \lambda |w_1|}_{f(w_1)}$$

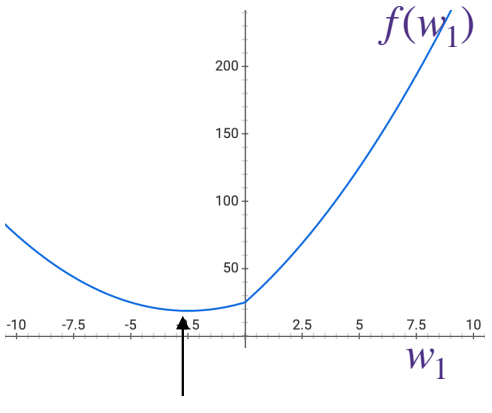
non-smooth

Convexity

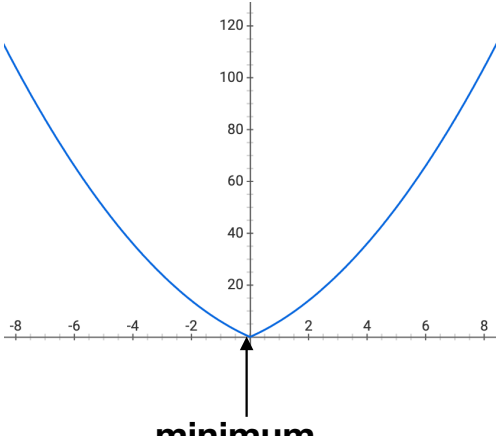
- to find the minimizer of $f(w_1)$, let's study some properties
- for simplicity, we represent the objective function as

$$f(w_1) = (aw_1 - b)^2 + \lambda |w_1|$$

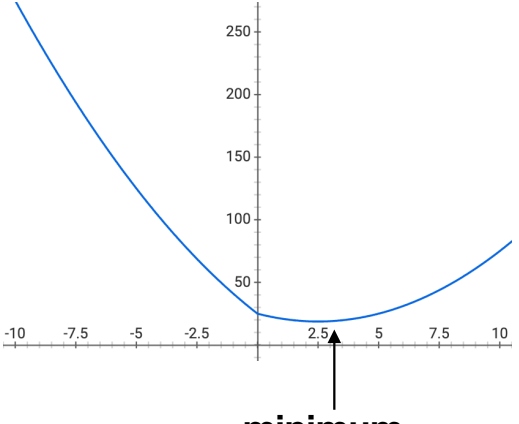
- this function is
  - **convex**, and
  - **non-differentiable**
- depending on the values of a, b, and λ, the function looks like one of the three below



minimum

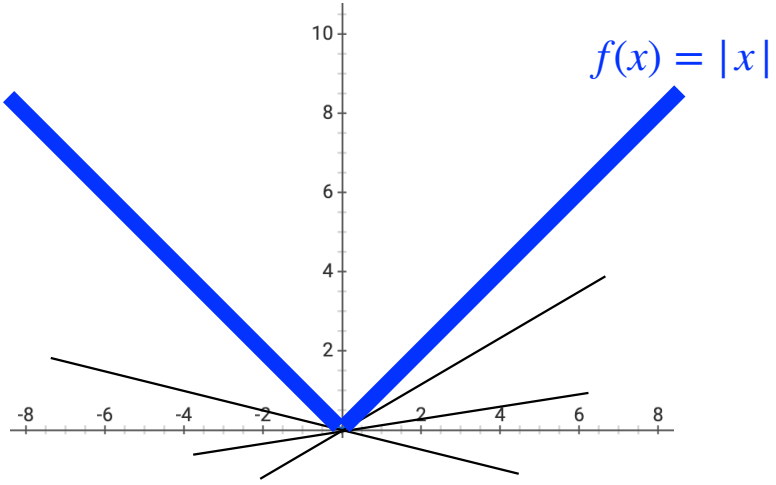


minimum



minimum

Convexity



- for a **non-differentiable** function, gradient is not defined at some points, for example at $x = 0$ for $f(x) = |x|$
- at such points, **sub-gradient** plays the role of gradient
  - sub-gradient at a differentiable point is the same as the gradient
  - sub-gradient at a non-differentiable point is a set of vector satisfying

$$\partial f(x) = \{ g \in \mathbb{R}^d \,|\, f(y) \geq f(x) + g^T(y - x), \text{ for all } y \in \mathbb{R}^d \}$$

  - for example, sub-gradient of $|\cdot|$ is $\partial |x| = \begin{cases} +1 & \text{for } x > 0 \\ [-1, 1] & \text{for } x = 0 \\ -1 & \text{for } x < 0 \end{cases}$

Computing the sub-gradient

$$w_1^{(t)} = \arg\min_{w_1 \in \mathbb{R}} \underbrace{\left\| \mathbf{X}[:,1] w_1 - \left(\mathbf{y} - \mathbf{X}[:,2:d] w_{2:d}^{(t-1)} \right) \right\|_2^2 + \lambda |w_1|}_{f(w_1)}$$

$$f(w_1) = (a w_1 - b)^2 + \lambda |w_1| + \text{constant}$$

Where $a = \sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}$, and

$$b = \frac{\mathbf{X}[:,1]^T (\mathbf{y} - \mathbf{X}[:,2:d] w_{2:d}^{(t-1)})}{\sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}}$$

$$\partial f(w_1) = 2a(aw_1 - b) + \lambda \partial |w_1|$$

$$= \begin{cases} 2a(aw_1 - b) + \lambda & w_1 > 0 \\ -2ab + \lambda[-1, 1] & w_1 = 0 \\ = [-2ab - \lambda, -2ab + \lambda] & \\ 2a(aw_1 - b) - \lambda & w_1 < 0 \end{cases}$$

Computing the sub-gradient

$$w_1^{(t)} = \arg\min_{w_1 \in \mathbb{R}} \underbrace{\left\| \mathbf{X}[:,1] w_1 - \left(\mathbf{y} - \mathbf{X}[:,2:d] w_{2:d}^{(t-1)} \right) \right\|_2^2 + \lambda |w_1|}_{f(w_1)}$$

- We have $f(w_1) = (aw_1 - b)^2 + \lambda |w_1|$ + constants, with
  - $a = \sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}$, and
  - $b = \dfrac{\mathbf{X}[:,1]^T \left(\mathbf{y} - \mathbf{X}[:,2:d] w_{2:d}^{(t-1)} \right)}{\sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}}$

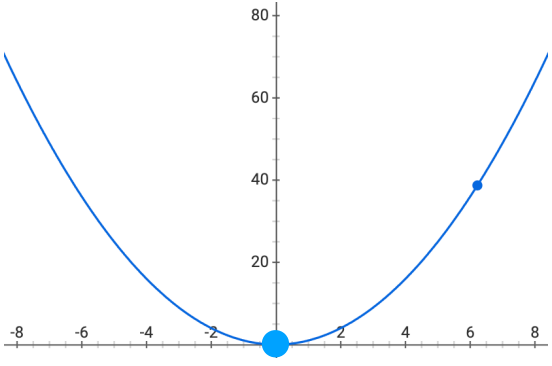
- $f(w_1)$ is non-differentiable, and its sub-gradient is

$$\partial f(w_1) = (2a(aw_1 - b) + \lambda \, \partial |w_1|$$

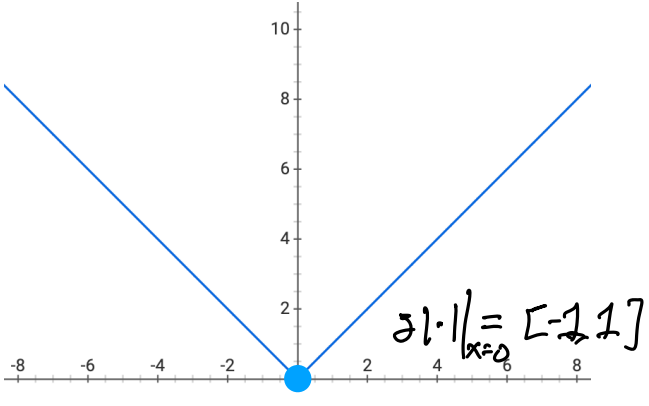
$$= \begin{cases} 2a(aw_1 - b) + \lambda & \text{for } w_1 > 0 \\ [-2ab - \lambda, -2ab + \lambda] & \text{for } w_1 = 0 \\ 2a(aw_1 - b) - \lambda & \text{for } w_1 < 0 \end{cases}$$

How do we find the minima?

- for **convex differentiable** functions, the minimum is achieved at points where gradient is zero



- for **convex non-differentiable** functions, the minimum is achieved at points where sub-gradient includes zero

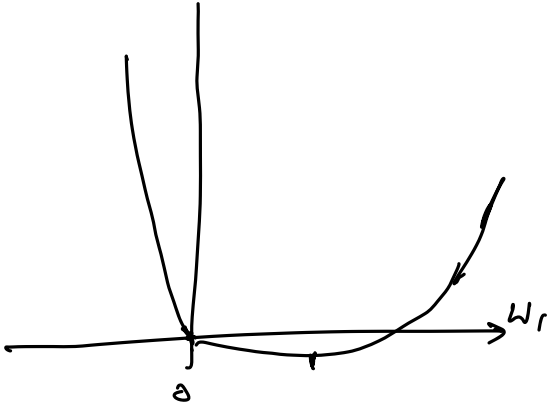


Computing the sub-gradient for $(aw_1 - b)^2 + \lambda |w_1|$

- the minimizer $w_1^{(t)}$ is when zero is included in the sub-gradient

$$\partial f(w_1) = \begin{cases} 2a(aw_1 - b) + \lambda & \text{for } w_1 > 0 \\ [-2ab - \lambda, -2ab + \lambda] & \text{for } w_1 = 0 \\ 2a(aw_1 - b) - \lambda & \text{for } w_1 < 0 \end{cases}$$

Case 1: minimum $w_1 > 0$.



$2a(aw_1 - b) + \lambda = 0$ & $w_1 > 0$

$w_1 = \frac{2ab - \lambda}{2a^2} > 0.$

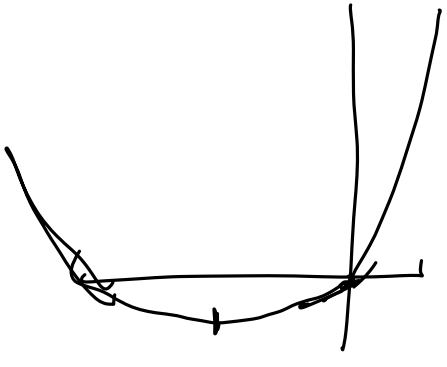
If $2ab - \lambda > 0$, then $w_1^{(t)} = \frac{2ab - \lambda}{2a^2}$

Computing the sub-gradient for $(aw_1 - b)^2 + \lambda |w_1|$

- the minimizer $w_1^{(t)}$ is when zero is included in the sub-gradient

$$\partial f(w_1) = \begin{cases} 2a(aw_1 - b) + \lambda & \text{for } w_1 > 0 \\ [-2ab - \lambda, -2ab + \lambda] & \text{for } w_1 = 0 \\ 2a(aw_1 - b) - \lambda & \text{for } w_1 < 0 \end{cases}$$

Case 2.



$2a(aw_1 - b) - \lambda = 0$, & $w_1 < 0$

$w_1 = \frac{2ab + \lambda}{2a^2} < 0.$

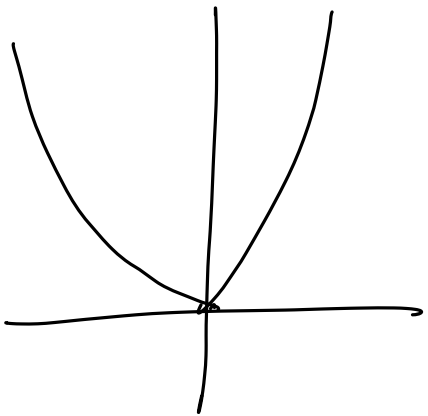
If $2ab + \lambda < 0$, then $w_1^{(t)} = \frac{2ab + \lambda}{2a^2}$

Computing the sub-gradient for $(aw_1 - b)^2 + \lambda |w_1|$

- the minimizer $w_1^{(t)}$ is when zero is included in the sub-gradient

$$\partial f(w_1) = \begin{cases} 2a(aw_1 - b) + \lambda & \text{for } w_1 > 0 \\ [-2ab - \lambda, -2ab + \lambda] & \text{for } w_1 = 0 \\ 2a(aw_1 - b) - \lambda & \text{for } w_1 < 0 \end{cases}$$

Case 3. $w_1 = 0$ if $-2ab - \lambda \leq 0 \leq -2ab + \lambda$

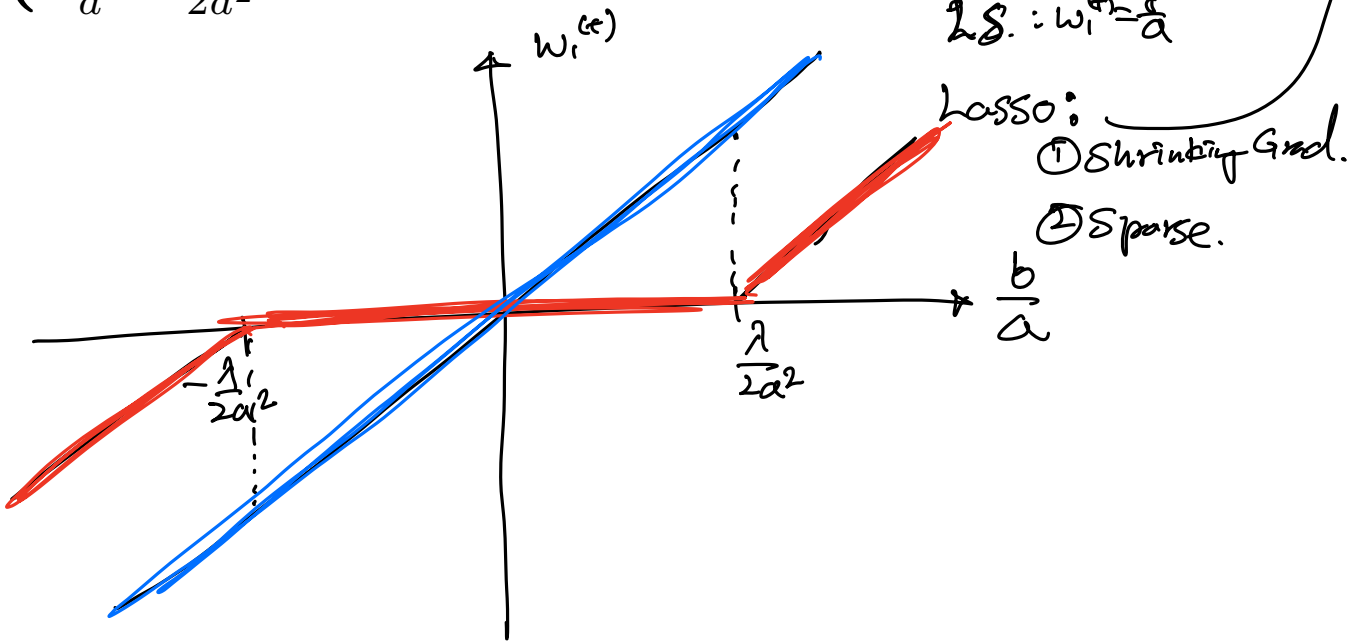


if $-\lambda \leq 2ab \leq \lambda$, then $w_1^{(t)} = 0$.

Computing the sub-gradient for $(aw_1 - b)^2 + \lambda |w_1|$

- considering all three cases, we get the following update rule by setting the sub-gradient to zero

$$w_1^{(t)} \leftarrow \begin{cases} \frac{b}{a} - \frac{\lambda}{2a^2} & \text{for } 2ab > \lambda \\ 0 & \text{for } -\lambda \leq 2ab \leq \lambda \\ \frac{b}{a} + \frac{\lambda}{2a^2} & \text{for } \lambda < -2ab \end{cases} \iff \frac{-\lambda}{2a^2} \leq \frac{b}{a} \leq \frac{\lambda}{2a^2}$$



LS.: $w_1^{(t)} = \frac{b}{a}$

Lasso:
① Shrinking Grad.
② Sparse.

How do we find the minimizer?

- the minimizer $w_1^{(t)}$ is when zero is included in the sub-gradient

$$\partial f(w_1) = \begin{cases} 2a(aw_1 - b) + \lambda & \text{for } w_1 > 0 \\ [-2ab - \lambda, -2ab + \lambda] & \text{for } w_1 = 0 \\ 2a(aw_1 - b) - \lambda & \text{for } w_1 < 0 \end{cases}$$

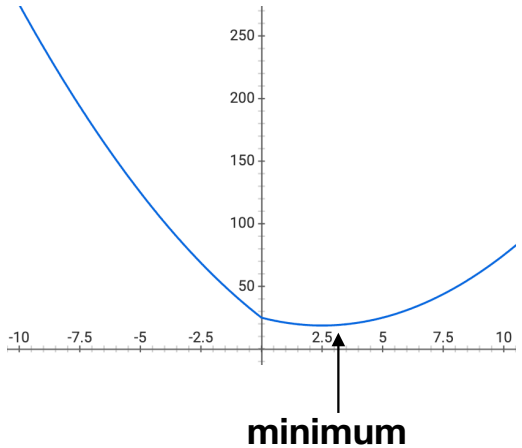
- case 1:
  - $2a(aw_1 - b) + \lambda = 0$ for some $w_1 > 0$
  - this happens when

$$w_1 = \frac{-\lambda + 2ab}{2a^2} > 0$$

  - hence,

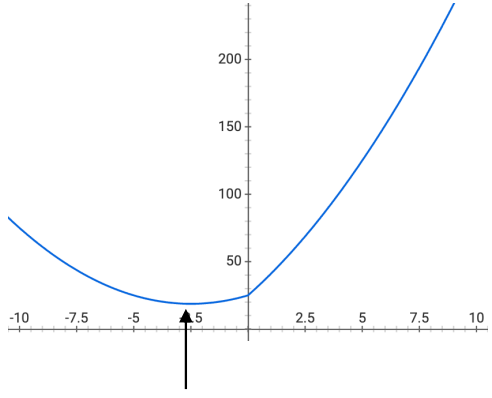
$$w_1^{(t)} \leftarrow \frac{b}{a} - \frac{\lambda}{2a^2},$$

  if $\lambda < 2ab$



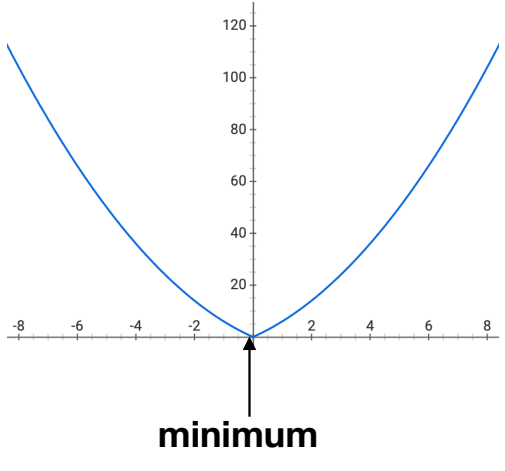
- case 2:
  - $2a(aw_1 - b) - \lambda = 0$ for some $w_1 < 0$
  - this happens when
    $$w_1 = \frac{\lambda + 2ab}{2a^2} < 0$$
  - hence,
    $$w_1^{(t)} \leftarrow \frac{b}{a} + \frac{\lambda}{2a^2},$$

    if $\lambda < -2ab$



- case 3:
  - $0 \in [-2ab - \lambda, -2ab + \lambda]$
  - and $w_1 = 0$
  - hence,
    $$w_1^{(t)} \leftarrow 0,$$

    if $-\lambda \leq 2ab \leq \lambda$

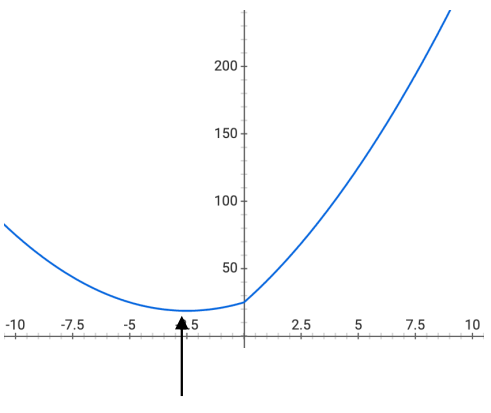


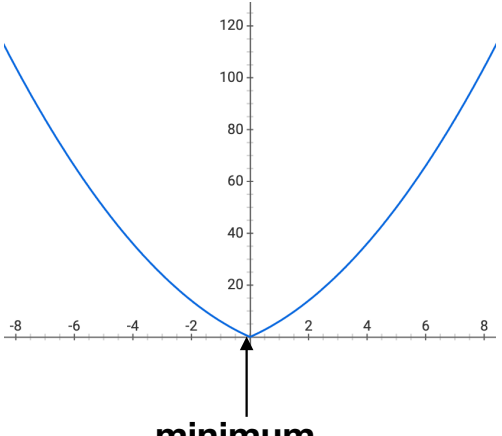
Coordinate descent on Lasso

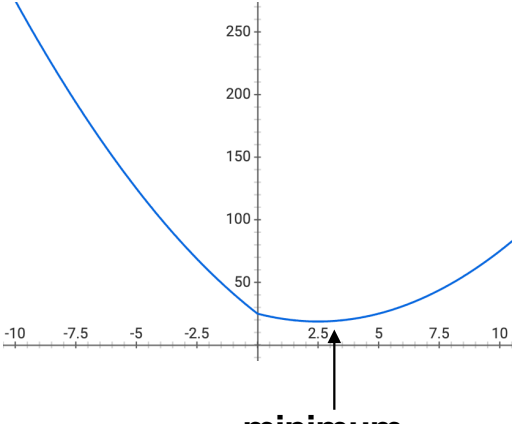
- considering all three cases, we get the following update rule by setting the sub-gradient to zero

$$w_1^{(t)} \leftarrow \begin{cases} \frac{b}{a} - \frac{\lambda}{2a^2} & \text{for } 2ab > \lambda \\ 0 & \text{for } -\lambda \leq 2ab \leq \lambda \\ \frac{b}{a} + \frac{\lambda}{2a^2} & \text{for } \lambda < -2ab \end{cases}$$

- where $a = \sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}$, and $b = \frac{\mathbf{X}[:,1]^T \left(\mathbf{y} - \mathbf{X}[:,2:d] w_{-1}\right)}{\sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}}$

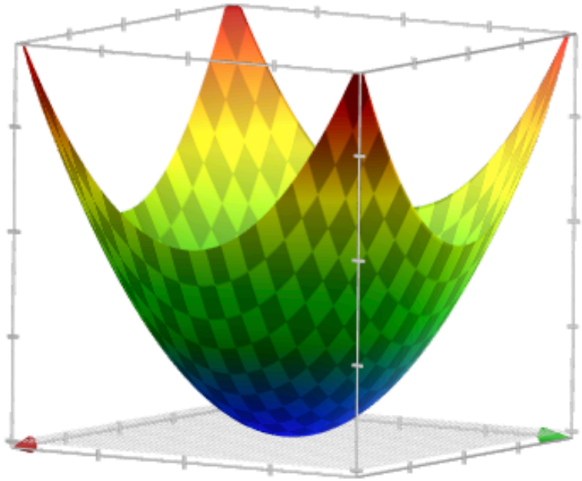


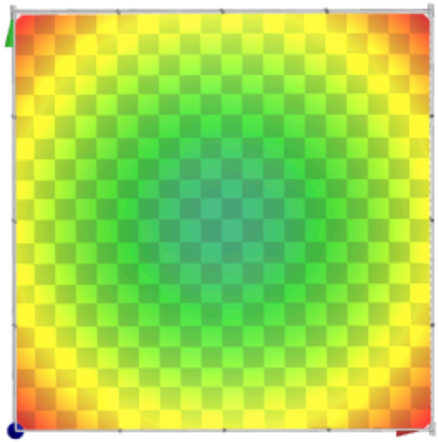




When does coordinate descent work?

- Consider minimizing a **differentiable convex** function $f(x)$, then coordinate descent converges to the global minima

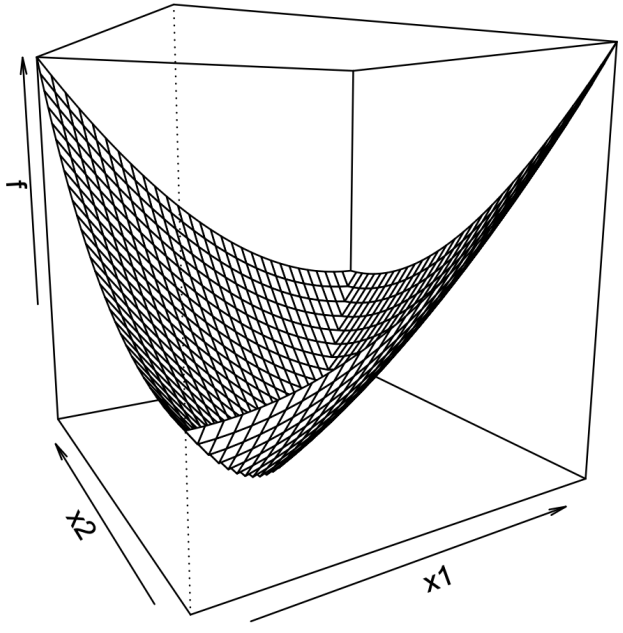


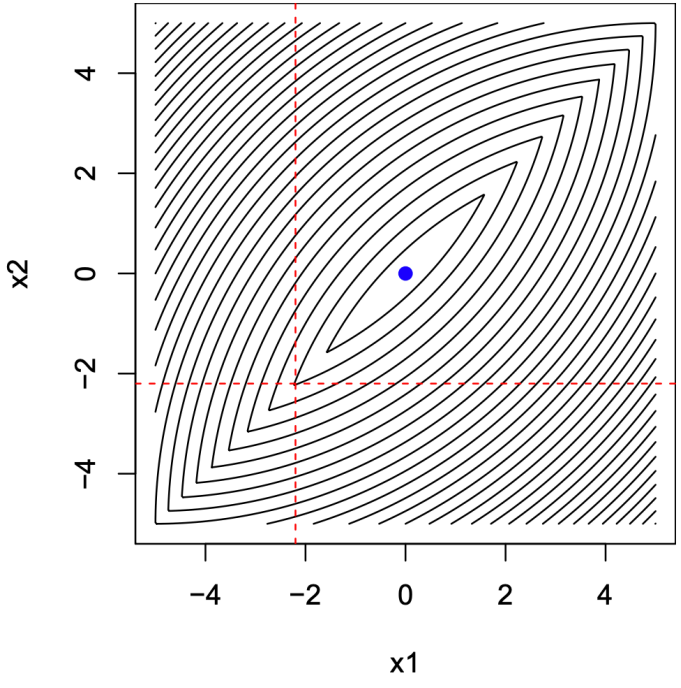


- when coordinate descent has stopped, that means $\frac{\partial f(x)}{\partial x_j} = 0$ for all $j \in \{1, \ldots, d\}$
- this implies that the gradient $\nabla_x f(x) = 0$, which happens only at minimum

When does coordinate descent work?

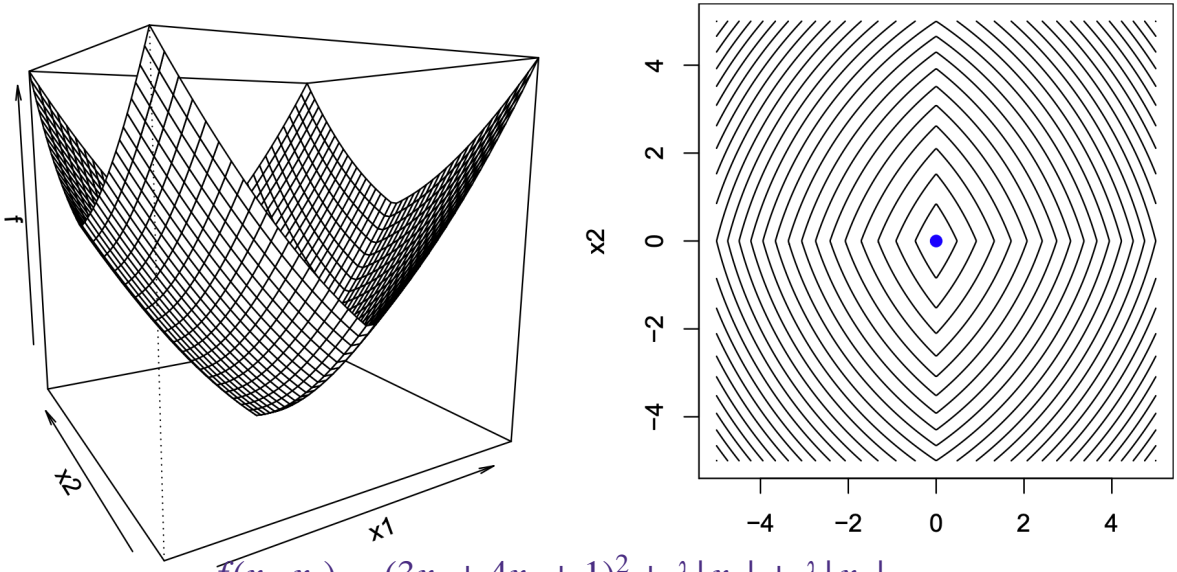
- Consider minimizing a **non-differentiable convex** function $f(x)$, then coordinate descent can get stuck





$$f(x_1, x_2) = (3x_1 + 4x_2 + 1)^2 + \lambda\,|x_1 - x_2|$$

When does coordinate descent work?

- then how can coordinate descent find optimal solution for Lasso?
- consider minimizing a **non-differentiable convex** function but has a structure of $f(x) = g(x) + \sum_{j=1}^{d} h_j(x_j)$, with differentiable convex function $g(x)$ and coordinate-wise non-differentiable convex functions $h_j(x_i)$'s, then coordinate descent converges to the global minima
- 

$f(x_1, x_2) = (3x_1 + 4x_2 + 1)^2 + \lambda |x_1| + \lambda |x_2|$

Questions?