Gradient Descent cannot be applied to Lasso : $\min_{w} \|y - Xw\|_2^2 + \underbrace{\lambda \|w\|_1}_{\text{non-smooth}}$

$\downarrow$

Coordinate Descent.

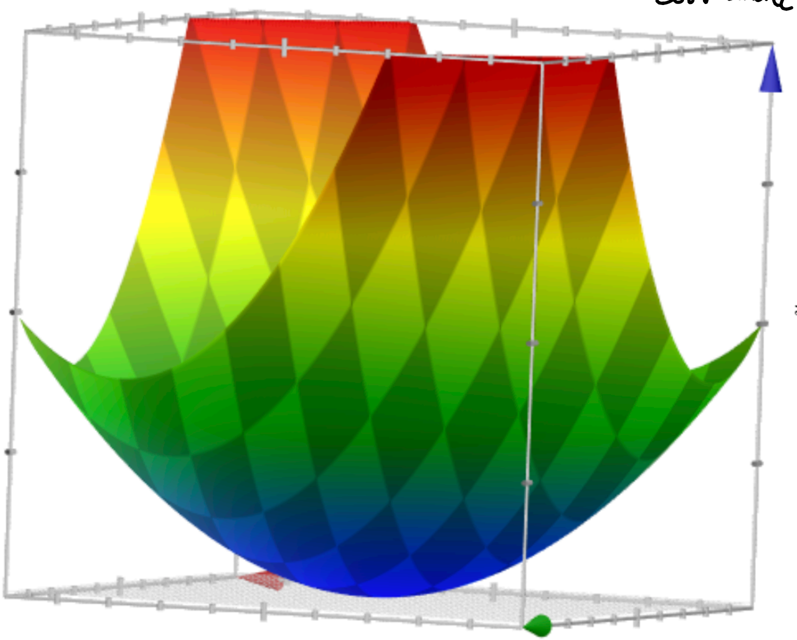
Coordinate Descent

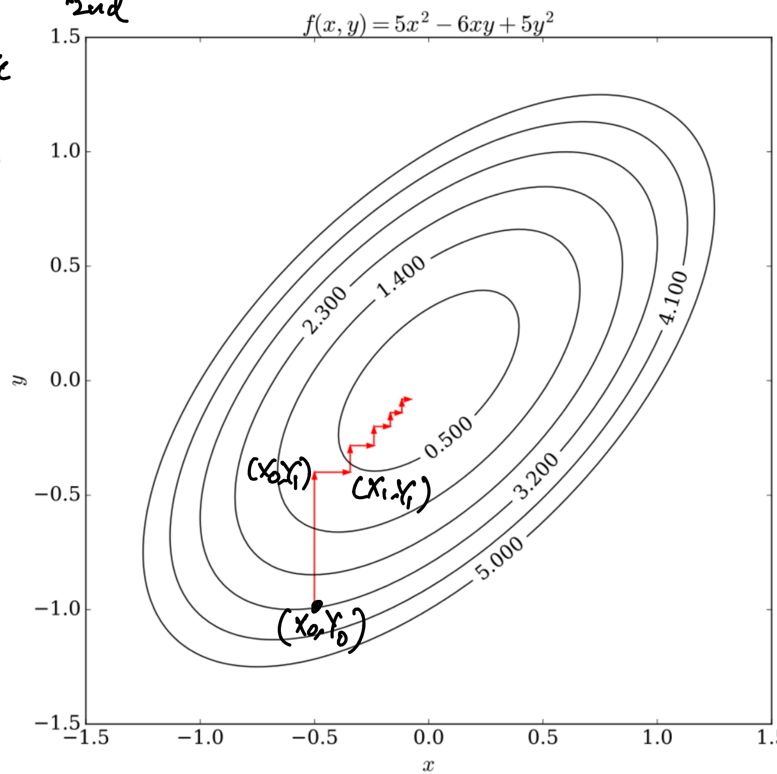
Optimization: how do we solve Lasso?

- 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 $f(x, y) = 5x^2 - 6xy + 5y^2$

1st coordinate

2nd





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 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

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

- let us derive an **analytical rule** for updating $w_j^{(t)}$

Coordinate descent for (un-regularized) linear regression

$j=1$ for example

$$\min_{w_1} \|Xw - y\|_2^2$$

$$= \left\| \begin{bmatrix} x_1 & x_{2:d} \end{bmatrix} \begin{bmatrix} w_1 \\ w_{2:d} \end{bmatrix} - \begin{bmatrix} y \end{bmatrix} \right\|_2^2 \longrightarrow (aw_1 - b)^2 + \text{Const}$$

$$= \| x_1 \cdot \underbrace{w_1}_{\in \mathbb{R}} - (y - x_{2:d} \cdot w_{2:d}) \|_2^2$$

$$= x_1^T x_1 \cdot w_1^2 - 2 x_1^T (y - x_{2:d} \cdot w_{2:d}) \cdot w_1 + \text{Const}$$

Define notation (a, b)

$$= (aw_1 - b)^2 + \text{Const}$$

$$a \triangleq \sqrt{x_1^T x_1}, \quad b \triangleq \frac{x_1^T (y - x_{2:d} w_{2:d})}{\sqrt{x_1^T x_1}}$$

$$w_1^* = \arg\min_{w_1 \in \mathbb{R}} (aw_1 - b)^2 + \text{Const} = \frac{b}{a} = \frac{x_1^T (y - x_{2:d} \cdot w_{2:d}^{(t)})}{x_1^T x_1} = w_1^{(t+1)}$$

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

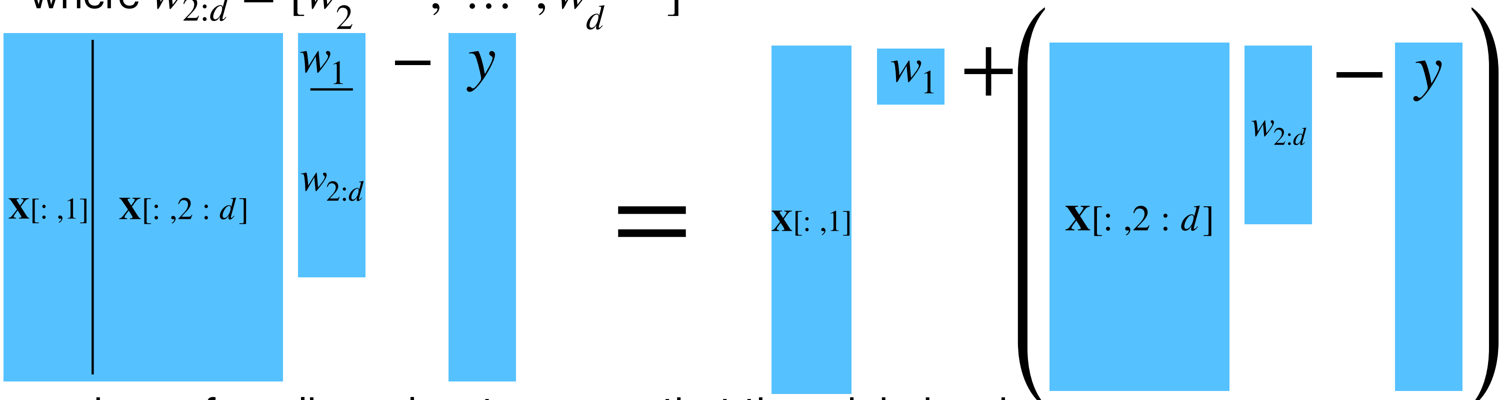
$$w_1^{(t)} \leftarrow \arg\min_{w_1} \|\mathbf{X}w - \mathbf{y}\|_2^2$$

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

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

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

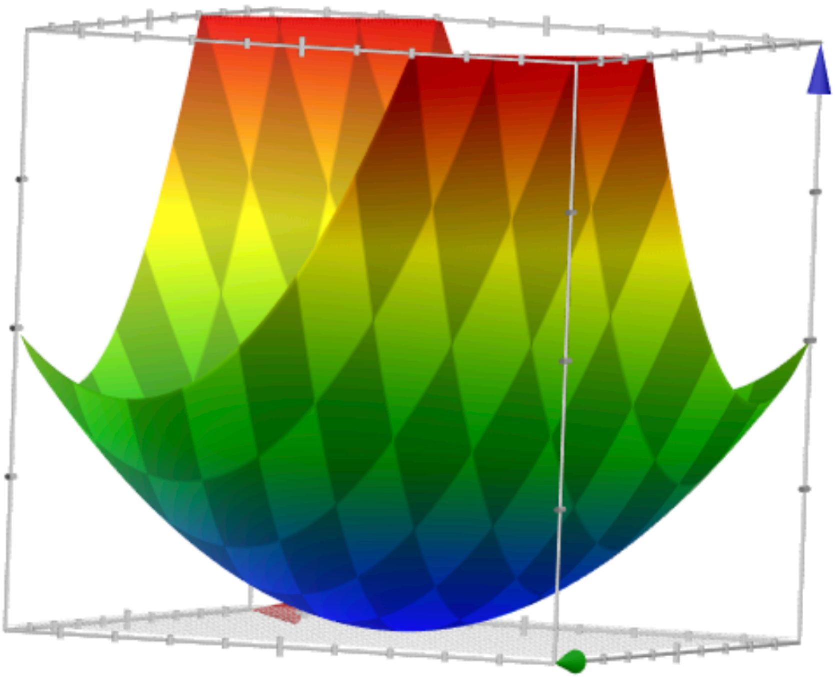
where $w_{2:d} = [w_2^{(t-1)}, \ \dots\ , w_d^{(t-1)}]^T$

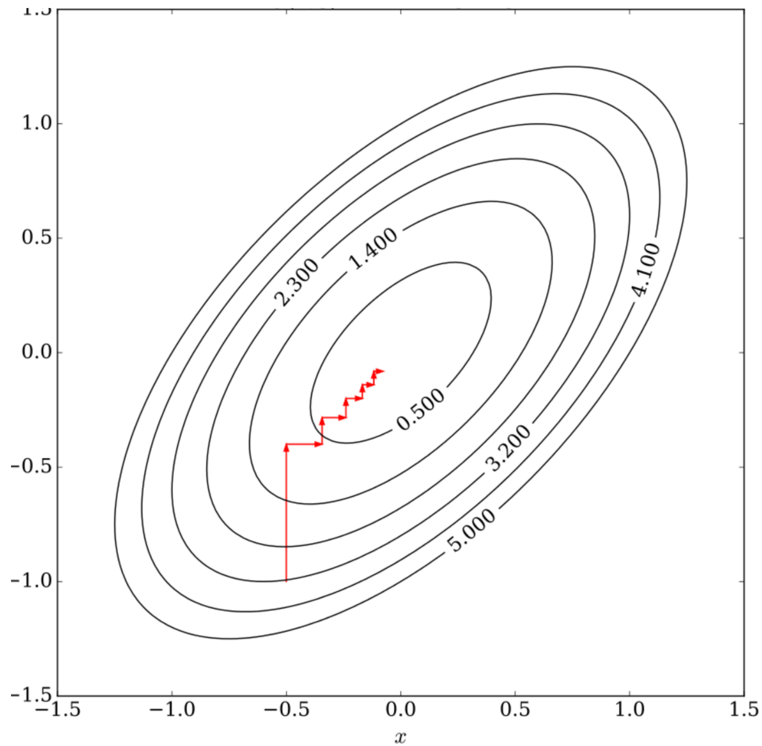


- we know from linear least squares that the minimizer is

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

- Coordinate descent applied to a quadratic loss





Coordinate descent for Lasso

- let us apply coordinate descent on Lasso, which minimizes
$$\text{minimize}_w \mathcal{L}(w) + \lambda \|w\|_1 = \|\mathbf{X}w - \mathbf{y}\|_2^2 + \lambda\, \|w\|_1$$

- 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

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

$$w_1^{(t+1)} \leftarrow \arg\min_{w_1} = (aw_1 - b)^2 + \lambda \cdot |w_1|$$

  - hence, the coordinate descent update boils down to

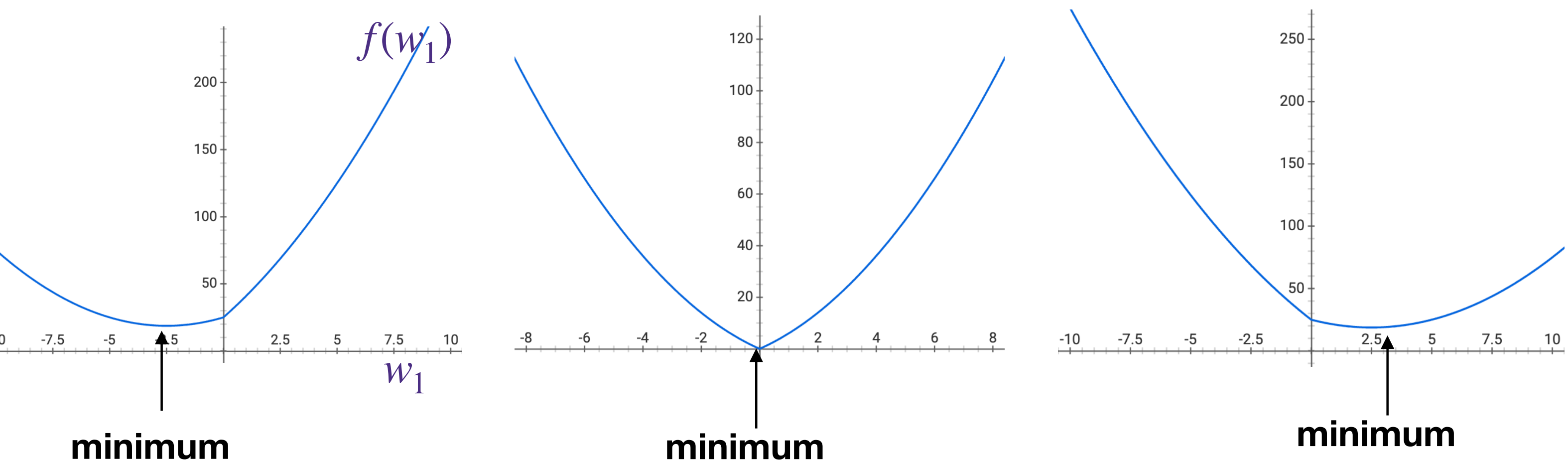
$$w_1^{(t)} \leftarrow \arg\min_{w_1} \underbrace{\left\| \mathbf{X}[:,1]w_1 - \left(\mathbf{y} - \mathbf{X}[:,2:d]w_{-2:d}\right) \right\|_2^2 + \lambda\, |w_1|}_{f(w_1)}$$

Convexity

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

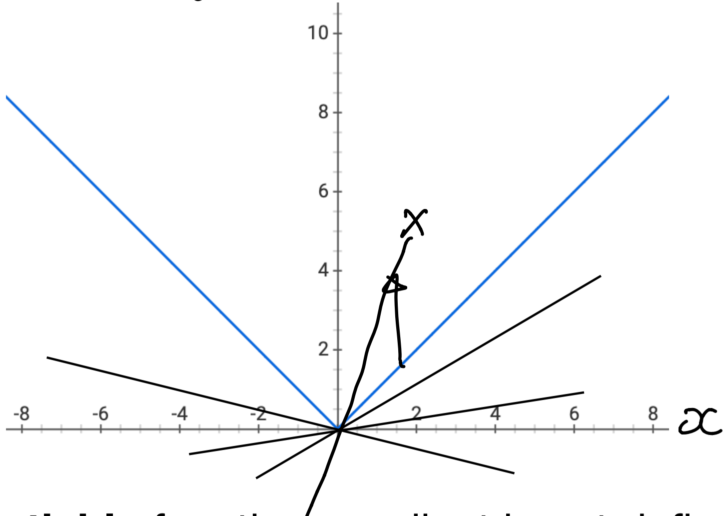
$$f(w_1) = (aw_1 - b)^2 + \lambda \, | \, w_1 \, |$$

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



Convexity

$f(x) = |x|$



- 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

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

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

Computing the sub-gradient

$$a = \sqrt{x_1^T x_1}$$

$$b = \frac{x_1^T (y - X_{2:d} w_{2:d}^{(t)})}{\sqrt{x_1^T x_1}}$$

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

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

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

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

$$[\,-2ab - \lambda,\ -2ab + \lambda\,]$$

Computing the sub-gradient

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

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

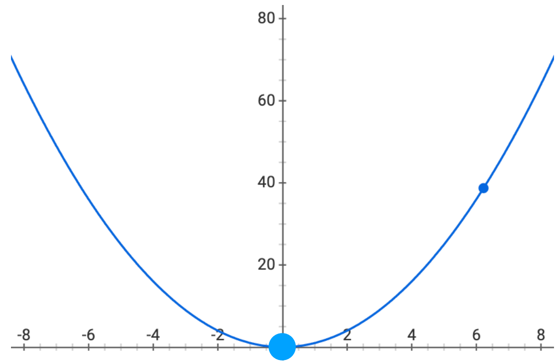
- $f(w_1)$ is non-differentiable, and its sub-gradient is

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

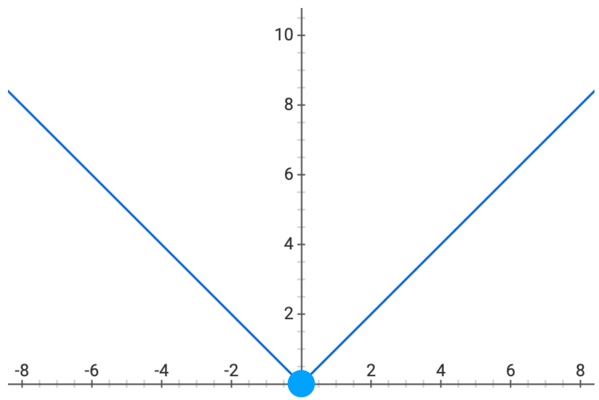
$$= \begin{cases} 2a(aw_1 - b) + \lambda & \text{for } w_1 > 0 \\ [-2ab - \lambda, -2ab + \lambda] & \text{for } w_1 = 0 \\ 2a(aw_1 - b) - \lambda & \text{for } w_1 < 0 \end{cases}$$

Convexity

- 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 -vector



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

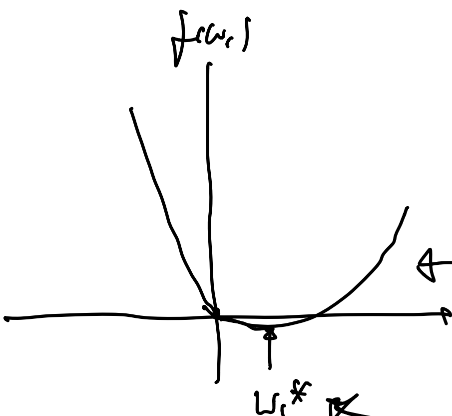
- the minimizer $w_1^{(t)}$ is when zero is included in the sub-gradient

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

Case 1. $w_1^* > 0$. $\longrightarrow$ $2a(aw_1^* - b) + \lambda = 0$

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

$\therefore$ If $2ab > \lambda$, then $w_1^* = \dfrac{2ab - \lambda}{2a^2}$



Computing the sub-gradient

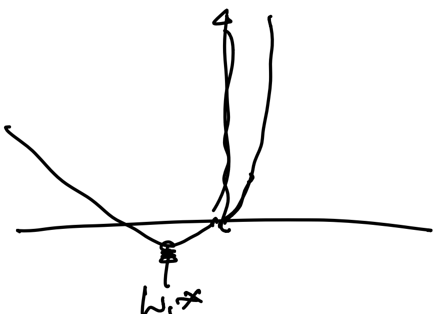
- the minimizer $w_1^{(t)}$ is when zero is included in the sub-gradient

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

Case 2: $w_1^* < 0$, $2a(aw_1^* - b) - \lambda = 0$

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

$\therefore$ If $2ab < -\lambda$, $w_1^* = \dfrac{2ab+\lambda}{2a^2}$

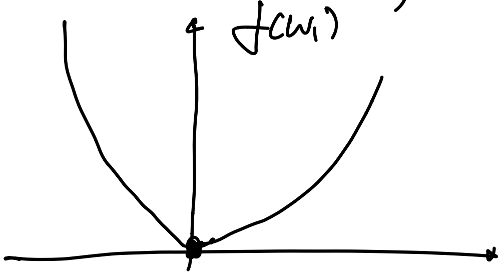


Computing the sub-gradient

- 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: $w_1^* = 0$,



$-2ab - \lambda \leq 0 \leq -2ab + \lambda$

$-\lambda \leq 2ab \leq \lambda. \longrightarrow w_1^* = 0.$

Computing the sub-gradient

update for linear regression

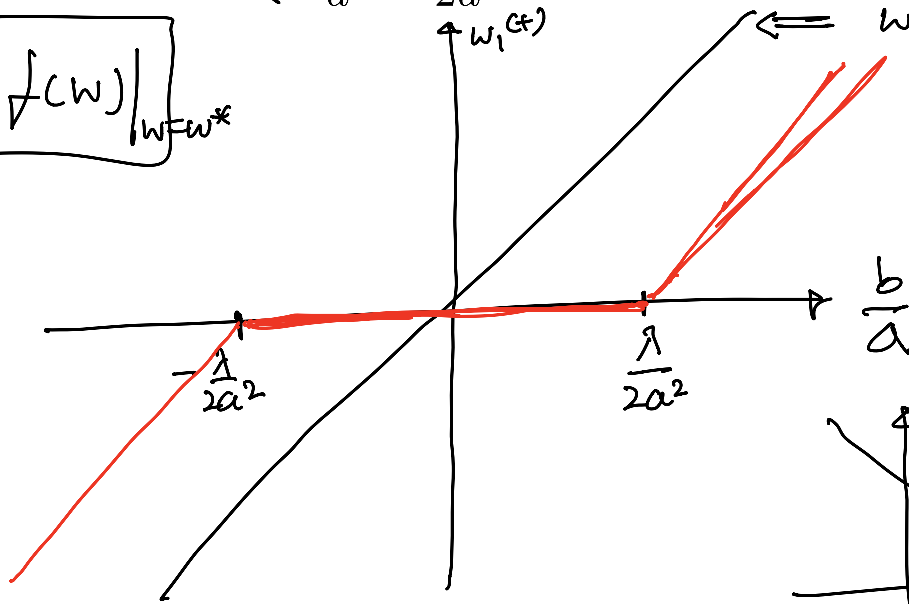
$w_1^{(t)} \leftarrow \frac{b}{a}$

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

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

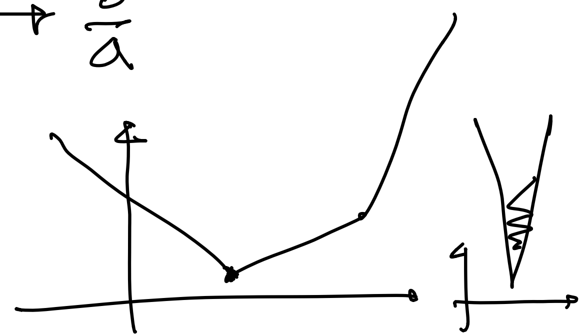
$\boxed{0 \in \partial f(w)\big|_{w=w^*}}$

$\Leftarrow w_1^{(t)} = \frac{b}{a}$ with no-regularization



① shrink.

② Sparsity.



How do we find the minimizer?

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

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

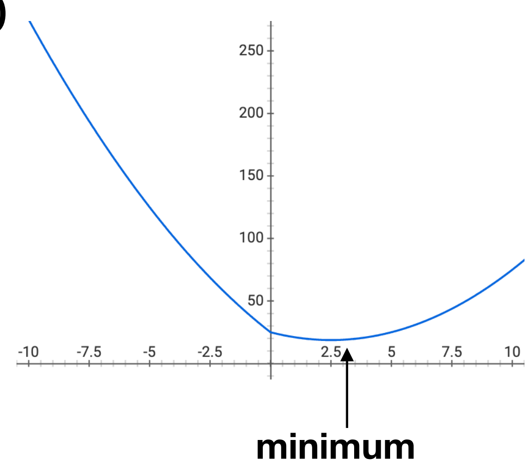
- case 1:
  - $2a(aw_1 - b) + \lambda = 0$ for some $w_1 > 0$
  - this happens when

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

  - hence,

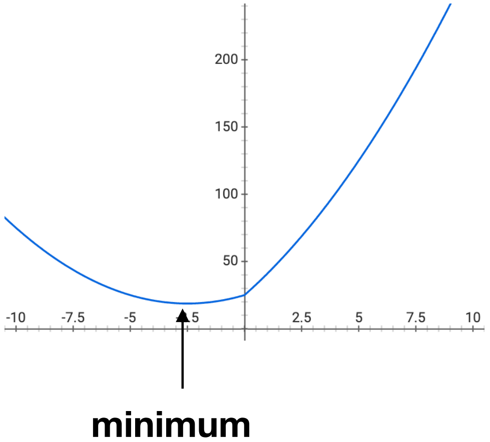
$$w_1^{(t)} \leftarrow \frac{b}{a} - \frac{\lambda}{2a^2},$$

if $\lambda < 2ab$



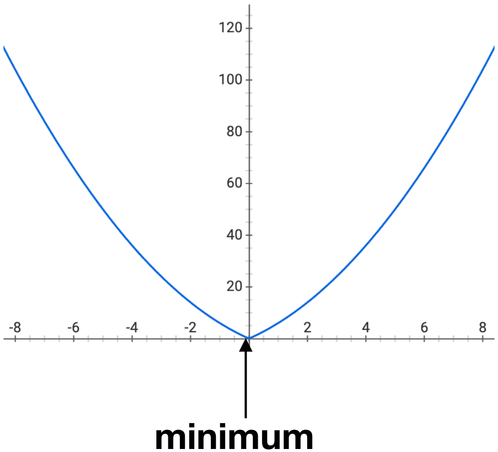
- case 2:
    - $2a(aw_1 - b) - \lambda = 0$ for some $w_1 < 0$
    - this happens when
      $$w_1 = \frac{\lambda + 2ab}{2a^2} < 0$$
    - hence,
      $$w_1^{(t)} \leftarrow \frac{b}{a} + \frac{\lambda}{2a^2},$$

      if $\lambda < -2ab$



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

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

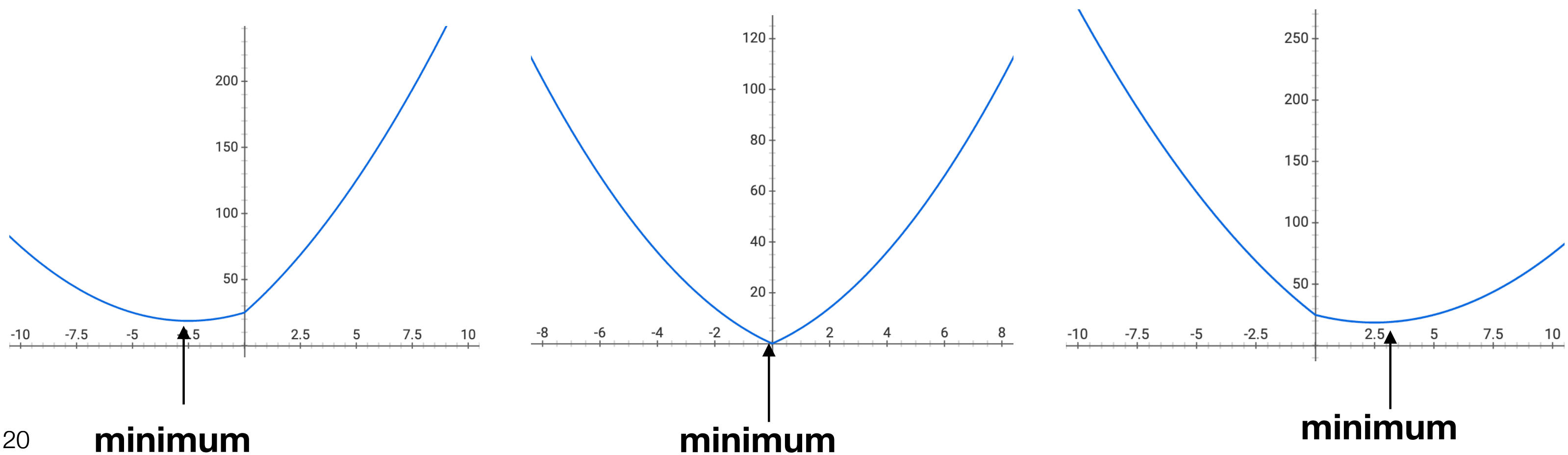


Coordinate descent on Lasso

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

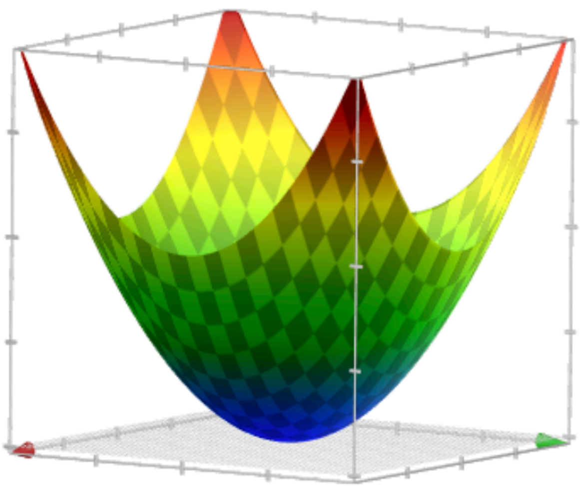
$$w_1^{(t)} \leftarrow \begin{cases} \frac{b}{a} - \frac{\lambda}{2a^2} & \text{for } 2ab > \lambda \\ 0 & \text{for } -\lambda \leq 2ab \leq \lambda \\ \frac{b}{a} + \frac{\lambda}{2a^2} & \text{for } \lambda < -2ab \end{cases}$$

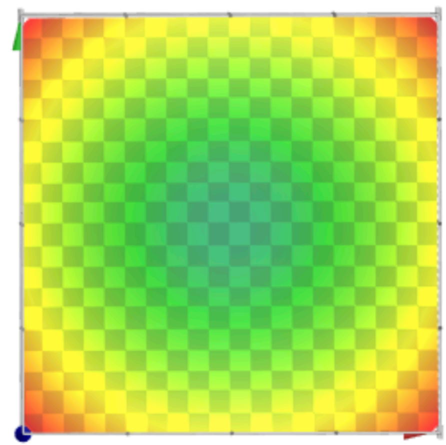
- where $a = \sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}$, and $b = \dfrac{\mathbf{X}[:,1]^T \big(\mathbf{y} - \mathbf{X}[:,2:d] w_{-1}\big)}{\sqrt{\mathbf{X}[:,1]^T \mathbf{X}[:,1]}}$



When does coordinate descent work?

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

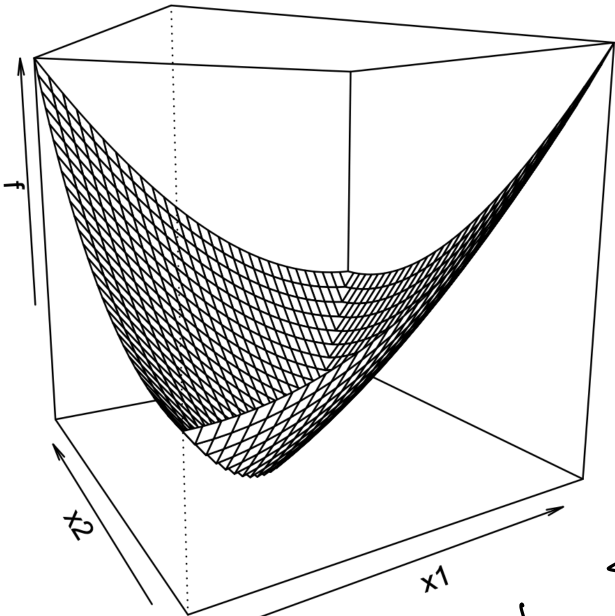


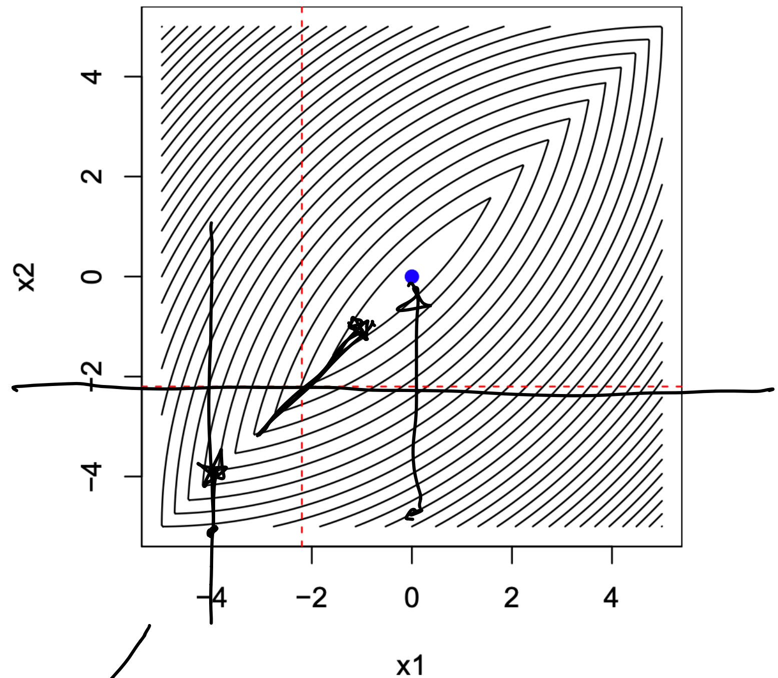


- when coordinate descent has stopped, that means
$\dfrac{\partial f(x)}{\partial x_j} = 0$ for all $j \in \{1, \dots, d\}$
- this implies that the gradient $\nabla_x f(x) = 0$, which happens only at minimum

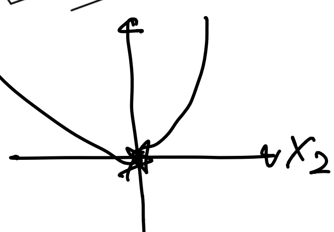
When does coordinate descent work?

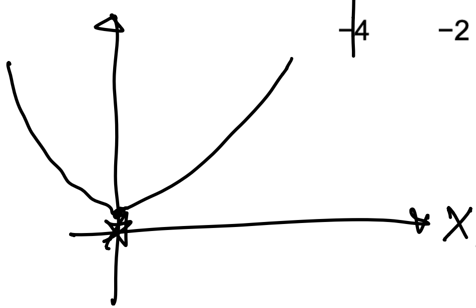
- Consider minimizing a **non-differentiable convex** function $f(x)$, then coordinate descent can get stuck





-





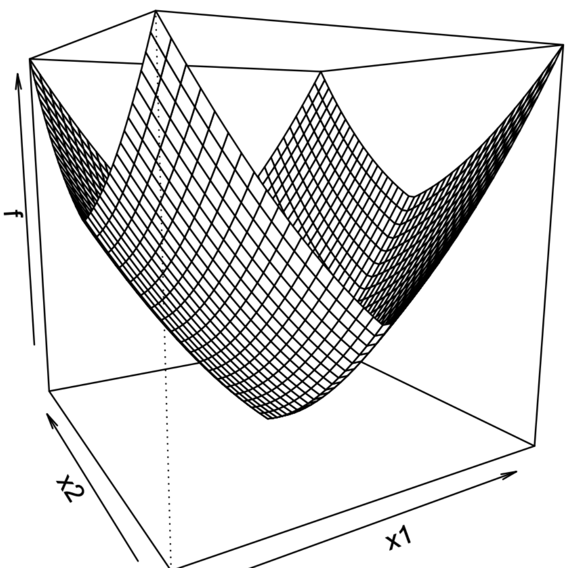
$\lambda_{max} \cong 1000$

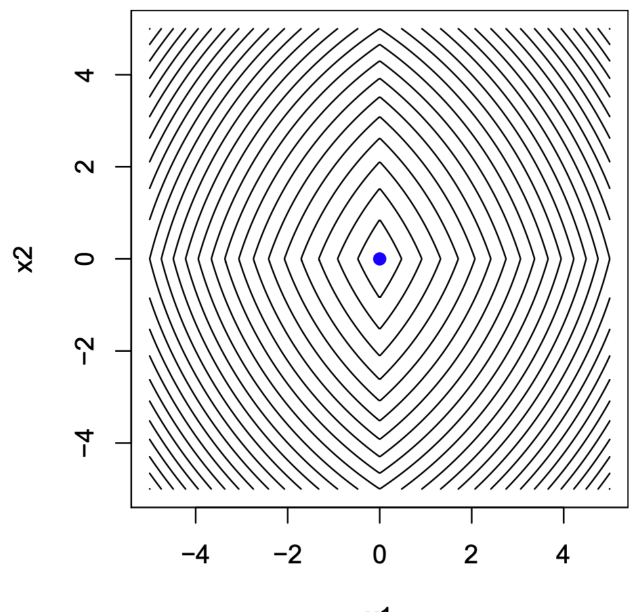
$\|Y - Xw\|^2 + \lambda \|w\|_1$

$\partial_w^2 \qquad \partial_w$

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 $\|w\|_1 = \sum_{j=1}^{d} |w_j|$ function $g(x)$ and coordinate-wise non-differentiable convex functions $h_j(x_j)$'s, then coordinate descent converges to the global minima





-

Stochastic Gradient Descent

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:** $\frac{1}{n}\sum_{i=1}^n \ell_i(w)$

Gradient Descent:

$$w_{t+1} = w_t - \eta \nabla_w \left(\frac{1}{n} \sum_{i=1}^n \ell_i(w) \right) \Big|_{w=w_t}$$

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:** $\frac{1}{n}\sum_{i=1}^n \ell_i(w)$

Gradient Descent:

$$w_{t+1} = w_t - \eta \nabla_w \left(\frac{1}{n} \sum_{i=1}^n \ell_i(w) \right) \Big|_{w=w_t}$$

Stochastic Gradient Descent:

$$w_{t+1} = w_t - \eta \nabla_w \ell_{I_t}(w) \Big|_{w=w_t}$$

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

$$\mathbb{E}[\nabla \ell_{I_t}(w)] =$$

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

$$\mathbb{E}[\ell(\bar{w}) - \ell(w_*)] \leq \frac{R}{2T\eta} + \frac{\eta G^2}{2} \leq \sqrt{\frac{RG^2}{T}} \qquad \eta = \sqrt{\frac{R}{G^2 T}}$$

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

(In practice use last iterate)

Stochastic Gradient Descent

Proof

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

Stochastic Gradient Descent

Proof

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

Stochastic Gradient Descent

Proof

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

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

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

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

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

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

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

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

Stochastic Gradient Descent

Proof

Jensen's inequality:
For any random $Z \in \mathbb{R}^d$ and convex function $\phi : \mathbb{R}^d \to \mathbb{R}$, $\phi(\mathbb{E}[Z]) \leq \mathbb{E}[\phi(Z)]$

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

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

Stochastic Gradient Descent

Proof

Jensen's inequality:
For any random $Z \in \mathbb{R}^d$ and convex function $\phi : \mathbb{R}^d \to \mathbb{R}$, $\phi(\mathbb{E}[Z]) \leq \mathbb{E}[\phi(Z)]$

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

$$\boxed{\mathbb{E}[\ell(\bar{w}) - \ell(w_*)] \leq \frac{R}{2T\eta} + \frac{\eta G}{2} \leq \sqrt{\frac{RG}{T}}} \qquad \eta = \sqrt{\frac{R}{GT}}$$

Mini-batch SGD

Instead of one iterate, average B stochastic gradient together

Advantages:

- Smaller variance
- Parallelization