CMSC 510 – L12
Regularization Methods for Machine Learning

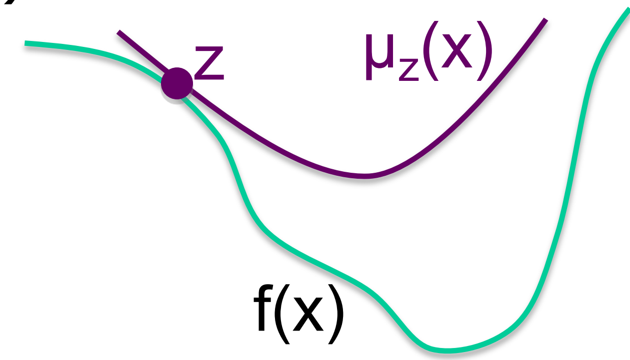
Instructor:

Dr. Tom Arodz

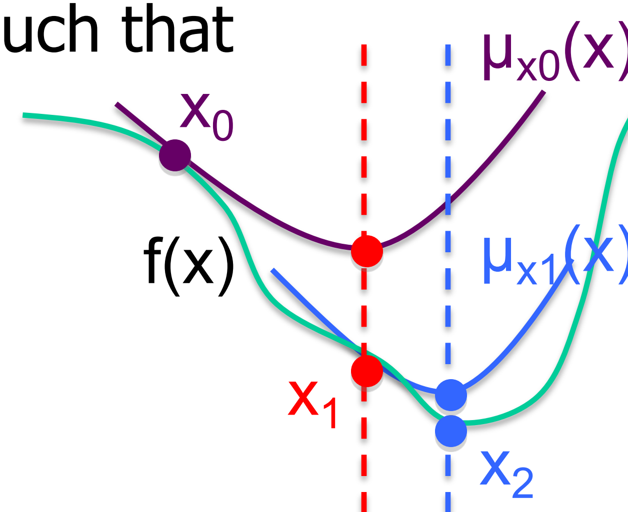
Recap: MM strategy

Majorization – minimization strategy

- Instead of directly minimizing function f(x)
- We design a family of "easier" functions μ_z such that:
  - $f(x) \leq \mu_z(x)$ for all x
  - $f(z) = \mu_z(z)$
  - μ_z is said to majorize function f(x) at z



- Iterative **majorization-minimization (MM)** procedure constructs a sequence $\{x_n\}$ such that $f(x_{n+1}) \leq f(x_n)$:
  - We start with arbitrary x_0
  - We construct $\mu_{x0}(x)$ and find its minimum x_1
  - We construct $\mu_{xn}(x)$ and find its minimum x_{n+1}

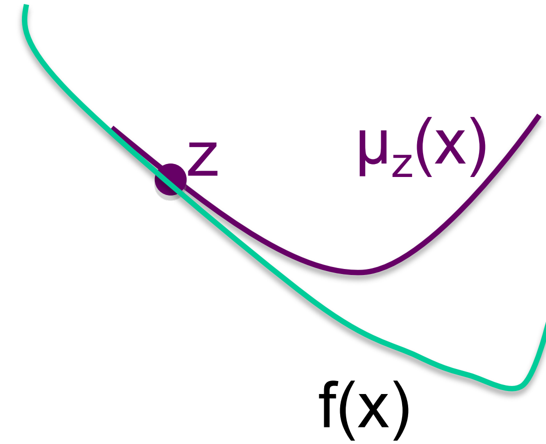


- We can show that $f(x_{n+1}) \leq f(x_n)$

Recap: MM strategy

For any f with L-Lipschitz gradient:

$$f(x) \le f(z) + \langle \nabla f(z), x-z \rangle + L/2 \, ||x - z||^2$$



Let $\mu_z(x) = f(z) + \langle \nabla f(z), x-z \rangle + L/2 \, ||x - z||^2$

μ_z majorizes f(x) at z

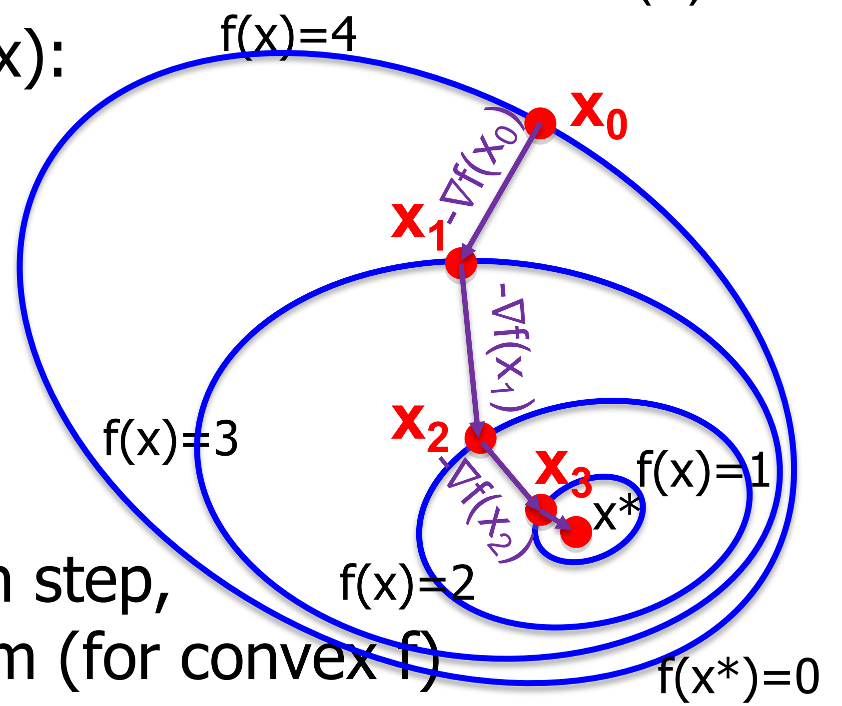
Minimum of $\mu_z(x)$ is $x = z - \nabla f(z)/L$

Let's apply the MM strategy using $\mu_z(x)$:

- We start from x_0
- We calculate $x_1 = x_0 - \nabla f(x_0)/L$
- We calculate $x_2 = x_1 - \nabla f(x_1)/L$
- $\mathbf{x_{n+1} = x_n - \nabla f(x_n)/L}$

We derived **gradient descent**!

We have a proof it goes down in each step, converging towards global minimum (for convex f)



Gradient descent

$<x,z+y>=<x,z>+<x,y>$

$<ax,by>=ab<x,y>$

$<x,y>=<y,x>$

$<x\ ,\ x> = ||x||^2$

Let: $\mu_z(x) = f(z) + <\nabla f(z),x-z> + L/2\ ||x - z\ ||^2$

Then: $\mu_z(x) = f(z) + L/2\ ||x - [z - \nabla f(z)/L]||^2 - 1/2L\ ||\nabla f(z)||^2$

Red is just another form of blue (let's denote $\nabla_z=\nabla f(z)$):

$$\mu_z(x)= f(z) + L/2\ ||\ x - [z - \nabla_z/L]\ \ ||^2 - 1/2L\ ||\ \nabla_z\ ||^2$$

$$= f(z) + L/2\ ||(x - z) + \nabla_z/L]||^2 - 1/2L\ ||\ \nabla_z\ ||^2$$

$$= f(z) + L/2 <(x - z) + \nabla_z/L, (x - z) + \nabla_z /L> - 1/2L < \nabla_z, \nabla_z >$$

$$= f(z) + L/2 \{ <x - z, x - z> + 2<\nabla_z/L,(x - z)> + <\nabla_z/L, \nabla_z/L> \} - L/2 <\nabla_z/L,\nabla/L>$$

$$= f(z) + L/2 <x - z, x - z> + <\nabla_z, (x - z)> + L/2<\nabla_z /L, \nabla_z /L> - L/2 <\nabla_z/L,\nabla_z/L>$$

$$= f(z) + L/2 <x - z, x - z> + <\nabla_z, (x - z)>$$

$$= f(z) + L/2\ ||x - z\ ||^2 + < \nabla_z, (x - z)>$$

$$= f(z) + <\nabla f(z), (x - z)> + L/2\ ||x - z\ ||^2 = \mu_z(x)$$

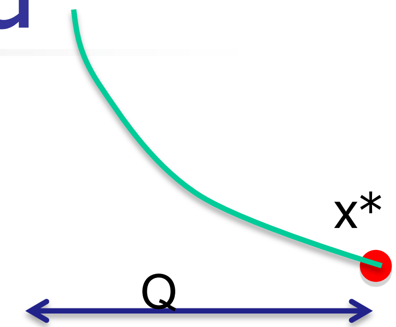
Now it's even easier to see that $x=z - \nabla f(z)/L$ is the minimum of $\mu_z(x)$

$$\mu_z(x) = f(z) + L/2\ ||x - [z - \nabla f(z)/L]||^2 - 1/2L\ ||\nabla f(z)||^2$$

Only the green part above depends on x, it's always non-negative, and we have

$$||\ [z - \nabla f(z)/L] - [z - \nabla f(z)/L]\ ||^2 = 0$$

Gradient projection method



Problem: minimize convex f(x)

s.t. x∈Q, where Q is a convex set

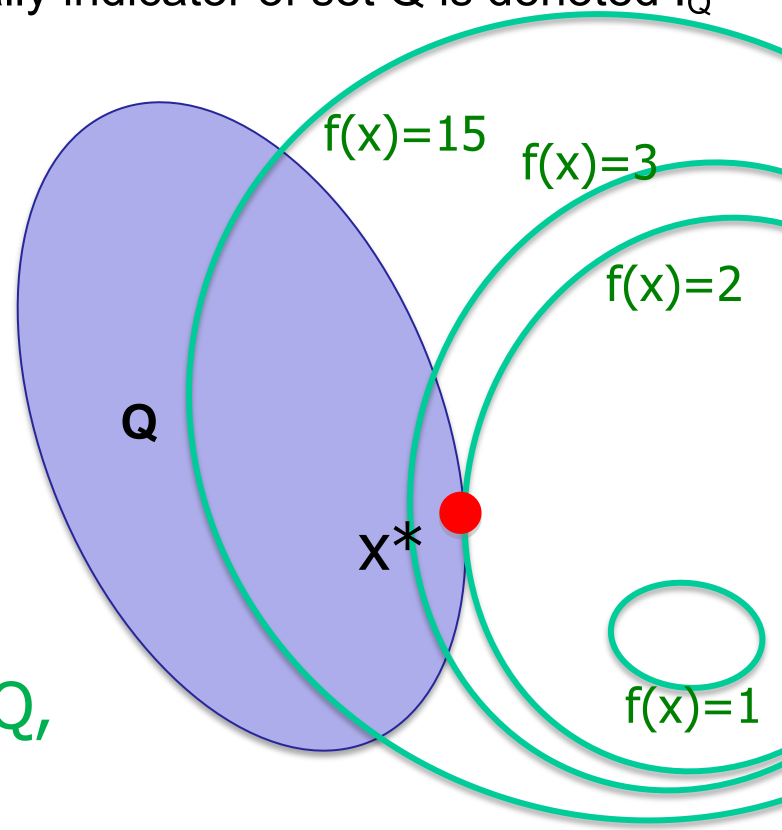
Let Q(x) be an *indicator function* for set Q

This is not the usual notation in literature; typically indicator of set Q is denoted I_Q

Q(x) = 0 if x∈Q,

Q(x) = ∞ otherwise

(has to be infinity; if finite large const., then Q(x) not convex)

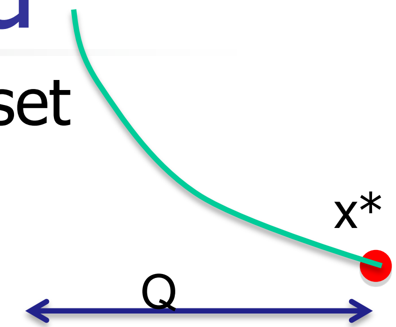


Equivalent problem:

minimize convex f(x)+Q(x)

f+Q has finite values only for x∈Q, so minimum is in Q

Gradient projection method



Problem: minimize convex $f(x)$ s.t. $x \in Q$, Q – convex set

Equivalent problem: minimize $f(x)+Q(x)$

Has finite values only for $x \in Q$, and also $Q(x)$ is convex

Majorizing function for f, assuming gradient of f is L-Lipschitz:

$$\mu_z(x) = f(z) + L/2\, ||x - [z - \nabla f(z)/L]||^2 - 1/2L\, ||\nabla f(z)||^2$$

$$f(x) \leq \mu_z(x) \qquad \text{for any } x$$

Still true if we add $Q(x)$ on both sides:

$$f(x) + Q(x) \leq \mu_z(x) + Q(x) \qquad \text{for any } x$$

Let $c_z = f(z) + 1/2L\, ||\nabla f(z)||^2$ *(a constant not depending on x)*

We get a **majorizing function** for **f(x) + Q(x)**:

$$f(x) + Q(x) \leq c_z + L/2\, ||x - [z - \nabla f(z)/L]||^2 + Q(x) \qquad \text{for any } x$$

Gradient projection method

Proximal operator: for function Q(x), positive b

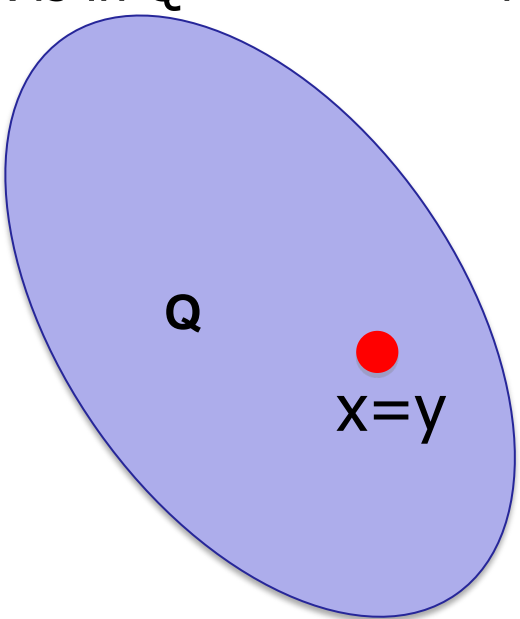
$$\mathbf{prox}_{Q,b}(y) = \text{argmin}_x\, Q(x)+b||x - y||^2$$

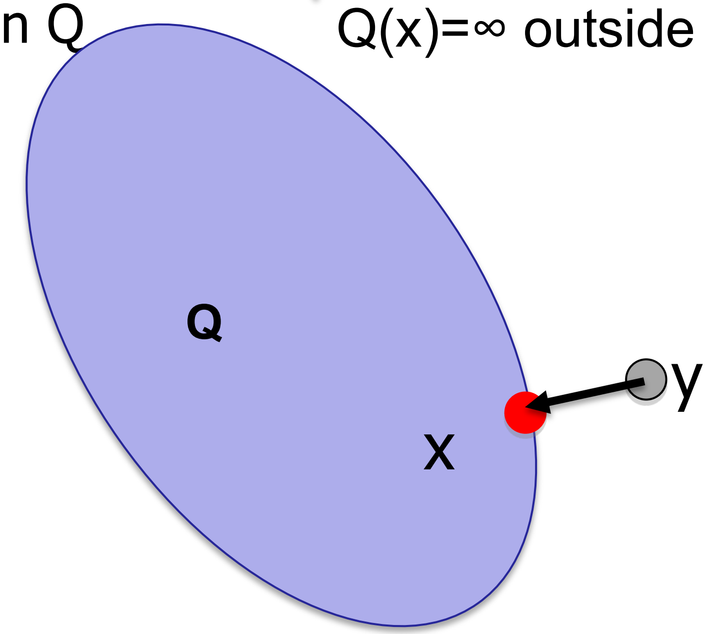
i.e., if Q(x) indicator of set Q, find x∈Q closest to the given y: $\mathbf{prox}_{Q,b}(y) = \text{argmin}_{x \in Q}\, b||x - y||^2$

If y in Q then x=y
so x is in Q

If y not in Q
we still get x that is in Q

because:
$Q(x)=\infty$ outside of Q





Gradient projection method

Problem: minimize f(x)+Q(x) where Q(x) - indicator f. of set Q

Majorizing function for f(x)+Q(x):

$$f(x) + Q(x) \le c_z + L/2\ ||x-[z - \nabla f(z)/L]||^2 + Q(x) \qquad \text{for any } x$$

MM iteration:

$$x_{n+1} = \operatorname{argmin}_x \quad Q(x) + L/2\ ||x - [x_n - \nabla f(x_n)/L]\ ||^2$$

Proximal operator: for a function Q(x)

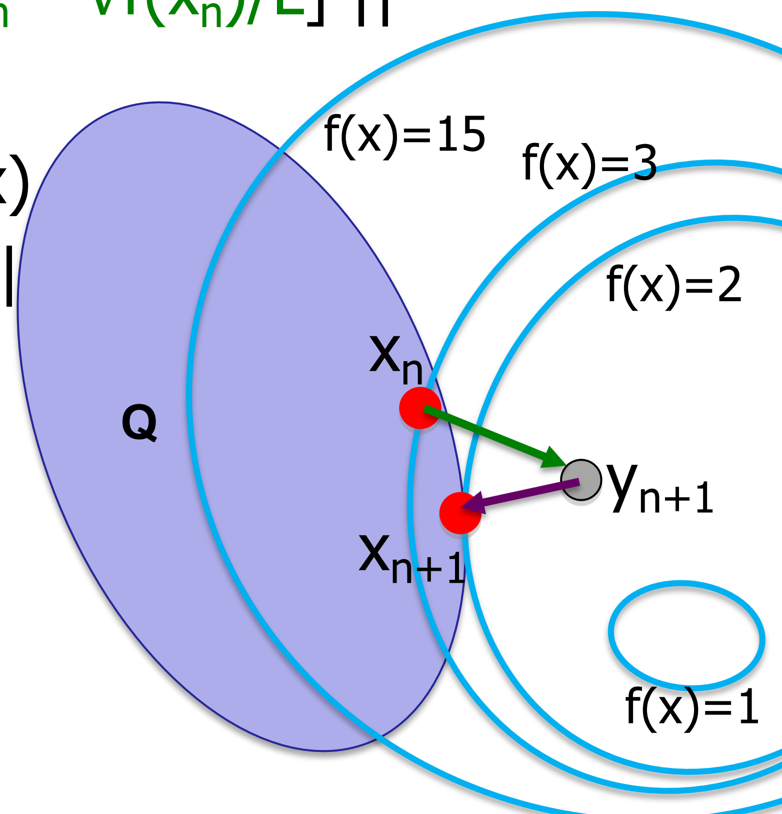
$$\mathbf{prox}_{Q,b}(y) = \operatorname{argmin}_x Q(x)+b||x - y||$$

So, MM iteration (in prox. notation):

1) $y_{n+1} = x_n - \nabla f(x_n)/L$
2) $x_{n+1} = \operatorname{prox}_{Q,L/2}(y_{n+1})$

same as:

$x_{n+1} = \operatorname{argmin}_x Q(x) + L/2\ ||x - y_{n+1}\ ||^2$



Summary: Gradient projection

Problem: minimize convex f(x) + Q(x)

Q(x) - indicator f. of a convex set Q

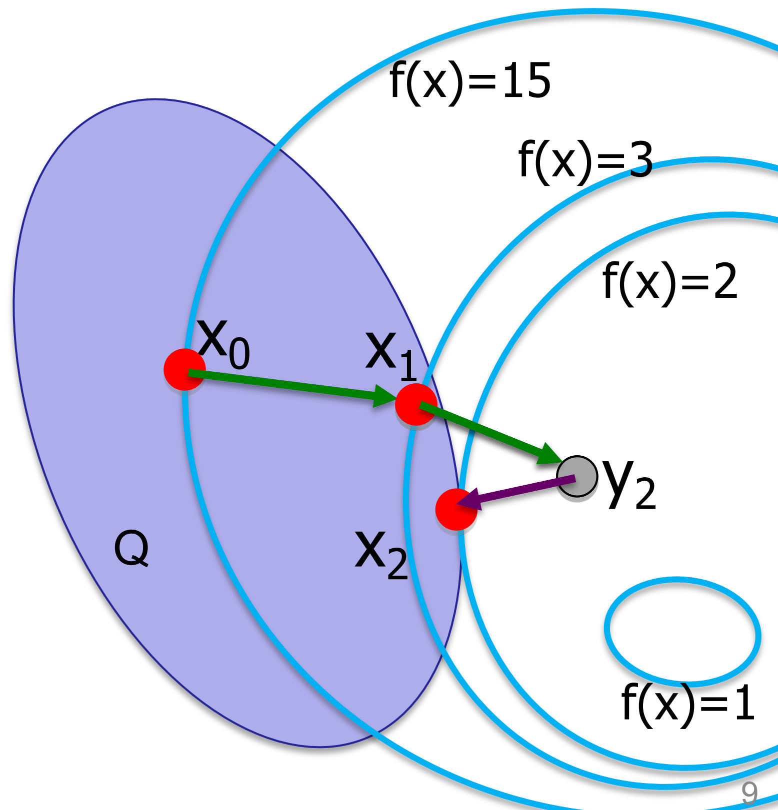
Proximal operator: for a convex function Q(x)

$$\mathbf{prox}_{Q,b}(y) = \mathrm{argmin}_x\, Q(x)+b||x - y||^2$$

MM iteration:

Gradient step: $y_{n+1} = x_n - \nabla f(x_n)/L$

Proximal step: $x_{n+1} = \mathrm{prox}_{Q,L/2}(y_{n+1})$



Guaranteed to converge:

it's proper MM, we're always going down, and there are no local minima

Is this approach limited to Q(x) representing convex sets Q?

Proximal gradient method

Problem: minimize convex f(x) + Q(x)

Proximal operator: $\mathrm{prox}_{Q,b}(y) = \mathrm{argmin}_x\, Q(x) + b||x - y||^2$

MM iteration:

Gradient step:

$y_{n+1} = x_n - \nabla f(x_n)/L$

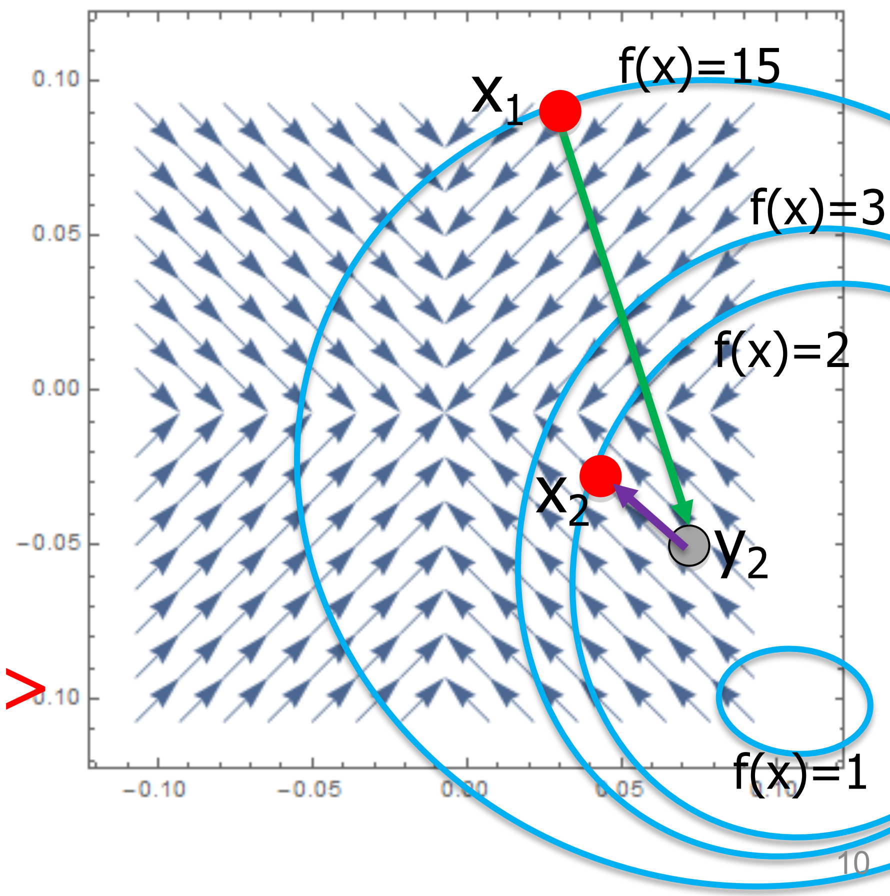
Proximal step:

$x_{n+1} = \mathrm{prox}_{Q,L/2}(y_{n+1})$

Derivation did not rely on Q(x) being indicator of a set

Only interpretation/plots did!

We can have different plots =>



Proximal gradient method

Problem: minimize convex $R(w) + \Omega(w)$

Proximal operator: $\mathrm{prox}_{\Omega,b}(v) = \mathrm{argmin}_w\, \Omega(w) + b\|w - v\|^2$

MM iteration:

Gradient step:

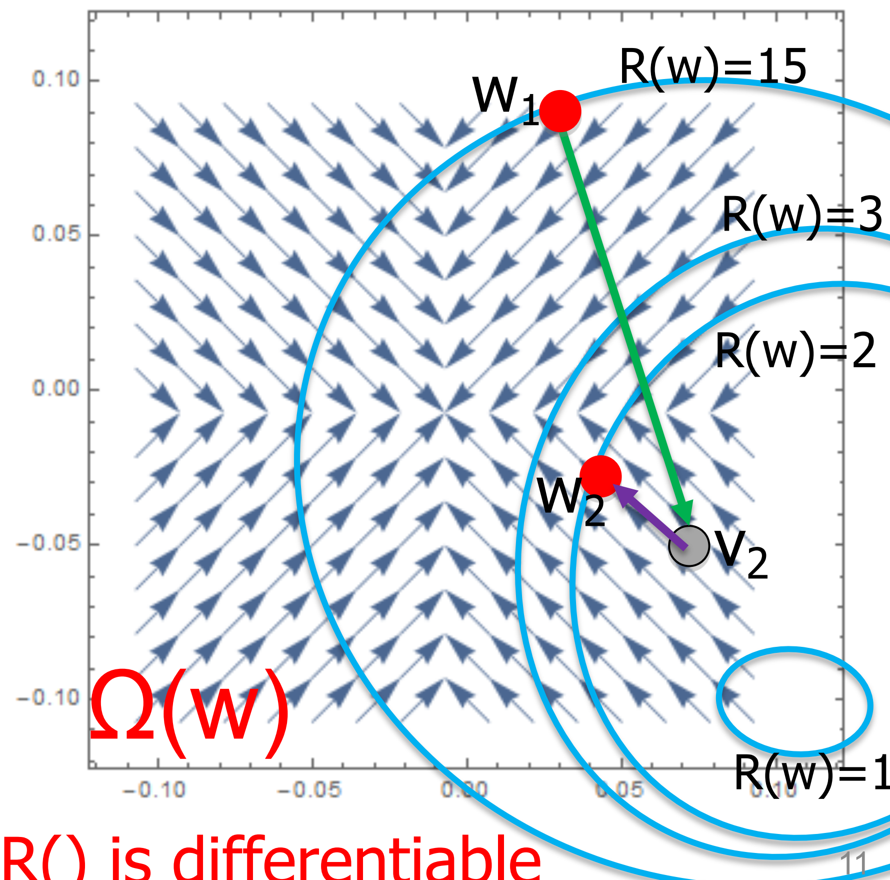
$$v_{n+1} = w_n - \nabla R(w_n)/L$$

Proximal step:

$$w_{n+1} = \mathrm{prox}_{\Omega,L/2}(v_{n+1})$$

MM converges to global minimum since $R + \Omega$ are convex

We didn't use gradient of Ω() only gradient of R()



Proximal gradient method:

Can be applied to non-differentiable Ω() as long as R() is differentiable

Proximal gradient method

Problem: minimize convex $R(w) + \Omega(w)$

Proximal operator: $\text{prox}_{\Omega,b}(v) = \text{argmin}_w\, \Omega(w)+b||w - v||^2$

MM iteration:

Gradient step:

$$v_{n+1}= w_n - \nabla R(w_n)/L$$

Proximal step:

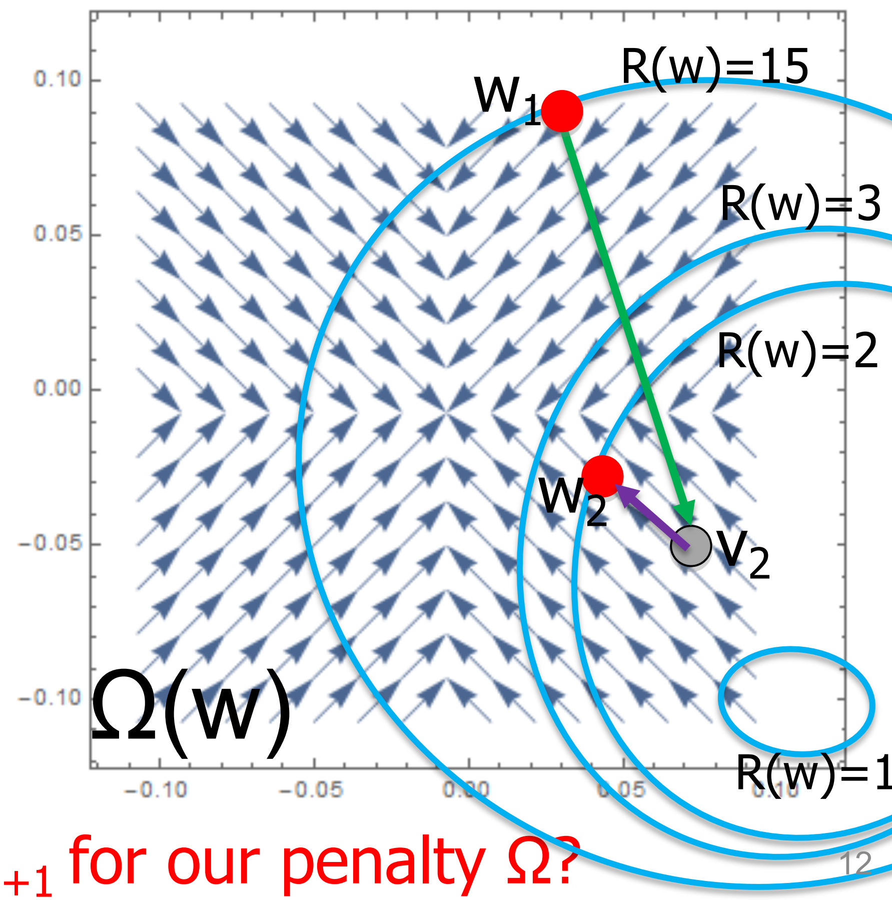
$$w_{n+1} = \text{prox}_{\Omega,L/2}(v_{n+1})$$

Proximal gradient method:

Can be applied to non-differentiable $\Omega()$ as long as $R()$ is differentiable

Key problem:

Can we solve the proximal step efficiently? Find w_{n+1} based on v_{n+1} for our penalty Ω?



Proximal operator for L_1 norm

Problem: minimize convex $R(w) + \Omega(w)$

Proximal operator: $\mathrm{prox}_{\Omega,b}(v) = \mathrm{argmin}_w\, \Omega(w)+b||w - v||^2$

Let's start with simple $\Omega(w)=||w||_1= \Sigma_f|w_f|$:

What is special about this $\Omega(w)$?

It's **separable**: $\Omega(w) =\Sigma_f\, \Omega_f(w_f)$ where $\Omega_f(\cdot)=|\cdot|$

- for separable functions Ω, proximal operator $\mathrm{prox}_{\Omega,b}(v)$ is easier to solve:

$$(\mathrm{prox}_{\Omega,b}(v))_f =\mathrm{prox}_{\Omega f,b}(v_f)$$

E.g.: if $\Omega(w_1,w_2)=\Omega 1(w_1)+\Omega 2(w_2)$
$\mathrm{prox}_{\Omega,b}(v) = (\mathrm{prox}_{\Omega 1,b}(v_1) , \mathrm{prox}_{\Omega 2,b}(v_2))$

  - prox($\cdot$) takes a vector as input, returns a vector on output
  - f-th coordinate of the result of prox(v) is prox() of f-th coordinate of input v

Why? because $||w - v||^2$ is separable:

$$||w - v||^2 = \Sigma_f (w_f - v_f)^2$$

- every coordinate/dimension *f* can be solved separately!

Proximal operator for L_1 norm

Proximal operator: $\text{prox}_{\Omega,b}(v) = \text{argmin}_w \, \Omega(w)+b||w - v||^2$

Let's start with simple $\Omega(w)=||w||_1= \Sigma_f|w_f|$:

$\Omega(w)$ separable, thus: ($\text{prox}_{\Omega,b}(v)$ $)_f$ $=\text{prox}_{\Omega f,b}(v_f)$

$\text{prox}_{\Omega f,b}(v_f) = \text{argmin}_{wf} \; |w_f| + b\,(w_f - v_f)^2$

How to solve $\arg\min_x |x| + b(x-z)^2$**, for a fixed z?**

convex, but non-differentiable: necessary and sufficient condition for global minimum is: $0 \in \partial\left(|x| + b(x-z)^2\right)$

$$-\frac{d\left(b(x-z)^2\right)}{dx} \in \partial|x|$$

$$-2b(x-z) \in G(x)$$

$$G(x) = \begin{cases} [-1,1], & \text{if } x = 0 \\ \{\text{sign}(x)\}, & \text{if } x \neq 0 \end{cases}$$

$$-2b(x-z) \in [-1,1] \text{ if } x=0 \implies x=0 \text{ if } 2bz \in [-1,1]$$

$$-2b(x-z) = 1 \text{ if } x>0 \implies 0<x=-\frac{1}{2b}+z \implies x=-\frac{1}{2b}+z \text{ if } z>\frac{1}{2b}$$

$$-2b(x-z) = -1 \text{ if } x<0 \implies 0>x=\frac{1}{2b}+z \implies x=\frac{1}{2b}+z \text{ if } z<-\frac{1}{2b}$$

Proximal operator for L_1 norm

Proximal operator for separable norm L_1, for single dimension:

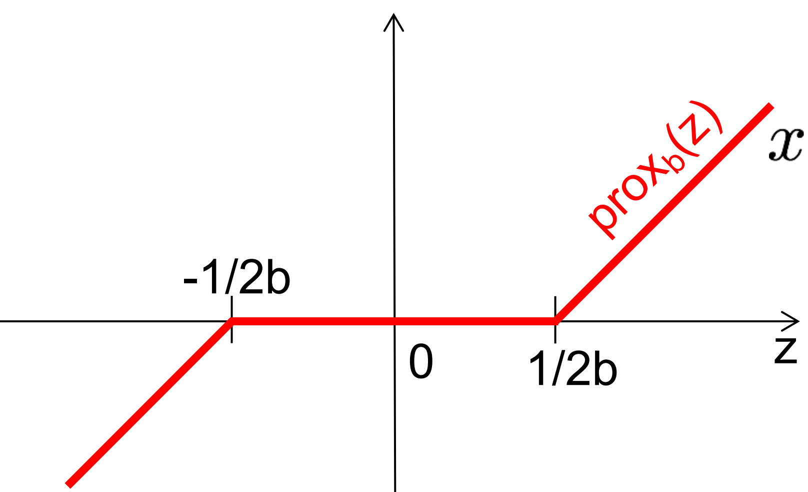
$$x^* = \text{prox}_b(z) = \arg\min_x |x| + b(x-z)^2$$

$$-2b(x-z) \in G(x) \qquad G(x) = \begin{cases} [-1,1], & \text{if } x = 0 \\ \{\text{sign}(x)\}, & \text{if } x \neq 0 \end{cases}$$

We have three cases for x:

$$-2b(x-z) \in [-1,1] \text{ if } x = 0 \implies x = 0 \text{ if } 2bz \in [-1,1]$$
$$-2b(x-z) = 1 \text{ if } x > 0 \implies 0 < x = -\tfrac{1}{2b} + z \implies x = -\tfrac{1}{2b} + z \text{ if } z > \tfrac{1}{2b}$$
$$-2b(x-z) = -1 \text{ if } x < 0 \implies 0 > x = \tfrac{1}{2b} + z \implies x = \tfrac{1}{2b} + z \text{ if } z < -\tfrac{1}{2b}$$

Thus, solution of $\text{prox}_b(z)$ for L_1 norm for each coordinate is:



$$x^* = \text{prox}_b(z) = \begin{cases} 0, & \text{if } |z| \leq \frac{1}{2b} \\ z - \frac{1}{2b}, & \text{if } z \geq \frac{1}{2b} \\ z + \frac{1}{2b}, & \text{if } z \leq -\frac{1}{2b} \end{cases}$$

$$x^* = \text{prox}_b(z) = \text{sign}(z)\left[|z| - \tfrac{1}{2b}\right]_+$$
$$[y]_+ = y \text{ if } y > 0 \text{ otherwise } 0$$

Proximal operator for L_1 norm

$$\Omega(w)=||w||_1= \Sigma_f|w_f|$$

Proximal operator: $\text{prox}_{\Omega,b}(v) = \text{argmin}_w\, \Omega(w)+b||w - v||^2$

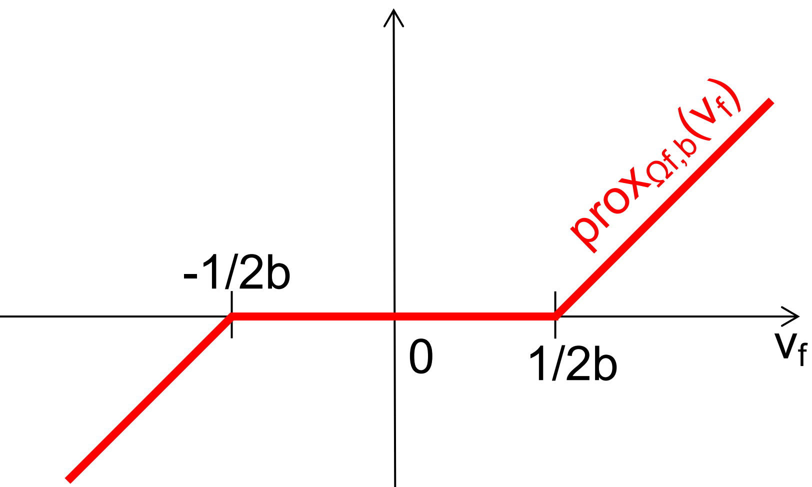
Separable $\Omega(w)$, so: $(\text{prox}_{\Omega,b}(v))_f = \text{prox}_{\Omega f,b}(v_f)$

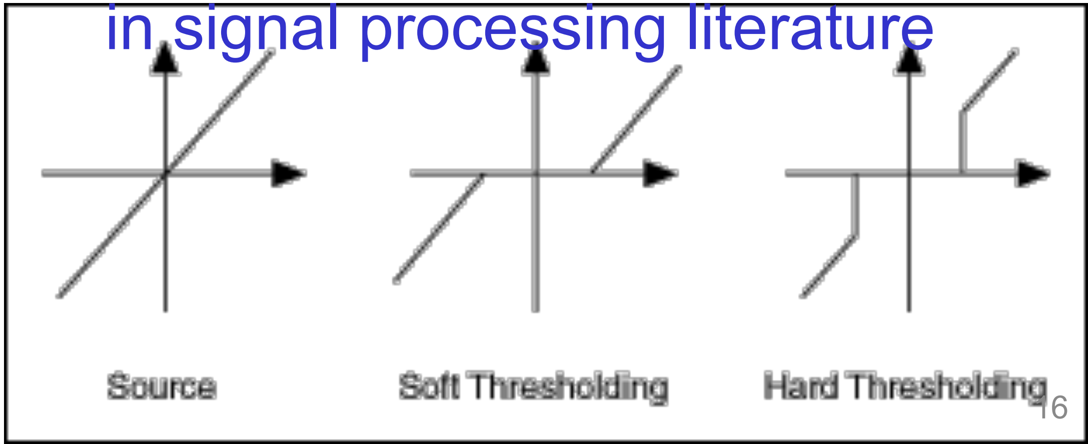
$$\text{prox}_{\Omega f,b}(v_f) = \text{argmin}_{wf}\ |w_f| + b\,(w_f - v_f)^2$$

Solution:

$$(\text{prox}_{\Omega,b}(v))_f = \text{prox}_{\Omega_f,b}(v_f) = \begin{cases} 0, & \text{if } |v_f| \le \frac{1}{2b} \\ v_f - \frac{1}{2b}, & \text{if } v_f \ge \frac{1}{2b} \\ v_f + \frac{1}{2b}, & \text{if } v_f \le -\frac{1}{2b} \end{cases}$$

This is called **soft thresholding** in signal processing literature





Solving L_1 norm regularized ERM

Problem: minimize L_1 regularized empirical risk: $R_S(w) + ||w||_1$

Gradient step: $v_{n+1} = w_n - \nabla R_S(w_n)/L$

Proximal step: $w_{n+1} = \text{prox}_{\Omega,L/2}(v_{n+1})$

Translates to an iterative algorithm for obtaining the global optimum w^* ($w^*=w_n$ for some large n):

$$(v_{n+1})_f = (w_n)_f - \frac{1}{L}\left.\frac{\partial \hat{R}_{S_m}(w)}{\partial w_f}\right|_{w=w_n}$$

$$(w_{n+1})_f = \begin{cases} 0, & \text{if } |(v_{n+1})_f| \leq \frac{1}{L} \\ (v_{n+1})_f - \frac{1}{L}, & \text{if } (v_{n+1})_f \geq \frac{1}{L} \\ (v_{n+1})_f + \frac{1}{L}, & \text{if } (v_{n+1})_f \leq -\frac{1}{L} \end{cases}$$

After some number of iterations, we'll have optimal w^*, that is, the optimal linear classifier $h(x)=w^{*T}x+w_0^*$

$(w_n)_f$ = f-th coordinate of vector w_n

Solving L_1 norm regularized ERM

Recall: $w_i^* = 0$ is part of minimum w^* if $\left|\frac{\partial \hat{R}_{S_m}(w^*)}{\partial w_i}\right| \leq 1$

Let's assume that at some iteration n, $(v_n)_f$ ended up near 0, so it got soft thresholded to 0: $(w_n)_f=0$

What happens to w_f in the future iterations n+1, n+2, … ?

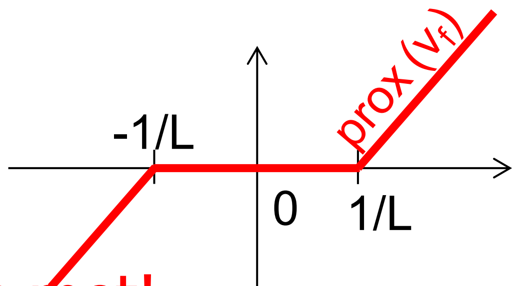
$$(v_{n+1})_f = (w_n)_f - \frac{1}{L}\left.\frac{\partial \hat{R}_{S_m}(w)}{\partial w_f}\right|_{w=w_n}$$

$$(w_{n+1})_f = \begin{cases} 0, & \text{if } |(v_{n+1})_f| \leq \frac{1}{L} \\ (v_{n+1})_f - \frac{1}{L}, & \text{if } (v_{n+1})_f \geq \frac{1}{L} \\ (v_{n+1})_f + \frac{1}{L}, & \text{if } (v_{n+1})_f \leq -\frac{1}{L} \end{cases} \quad (w_n)_f$$

If $(w_n)f=0$:

$$(v_{n+1})_f = -\frac{1}{L}\left.\frac{\partial \hat{R}_{S_m}(w)}{\partial w_f}\right|_{w=w_n}$$

$$(w_{n+1})_f = \begin{cases} 0, & \text{if } \left|\frac{1}{L}\left.\frac{\partial \hat{R}_{S_m}(w)}{\partial w_f}\right|_{w=w_n}\right| \leq \frac{1}{L} \\ \ldots \\ \ldots \end{cases}$$



w_f stays at 0 if the condition on gradient of risk is met!

$$(w_{n+1})_f = 0 \text{ if } \left|\left.\frac{\partial \hat{R}_{S_m}(w)}{\partial w_f}\right|_{w=w_n}\right| \leq 1$$

No surprise here, it has to, otherwise there's something wrong with our math

Proximal operator: Notation

We use notation:

$$\mathbf{prox}_{Q,b}(z) = \text{argmin}_x\, Q(x)+b||x - z||^2$$

in which soft thresholding is:

$$x^* = \text{prox}_b(z) = \text{sign}(z)\left[|z| - \frac{1}{2b}\right]_+$$

Often in literature we see slightly different notation:

$$\mathbf{prox}_{\lambda Q}(z) = \text{argmin}_x\, \lambda Q(x)+1/2\ ||x - z||^2$$

or

$$\mathbf{prox}_{Q,\lambda}(z) = \text{argmin}_x\, Q(x)+1/(2\lambda)\ ||x - z||^2$$

which make soft thresholding solution look simpler:

$$x^* = \text{prox}_\lambda(z) = \text{sign}(z)\left[|z| - \lambda\right]_+$$