CMSC 510 – L10
Regularization Methods for Machine Learning

Instructor:
Dr. Tom Arodz

Regularization – big picture

- Building an accurate ***Classifier***
- Key problem:
  - Do we have enough samples to describe all of variability of our two classes in the feature space?
  - **Rarely!**
  - Only recently huge datasets started to emerge
    - e.g. Google has pretty good overview of all images on internet
    - But still, not everything has been photographed and posted online
- **Regularization:**
  - Using a priori knowledge (not just training data) to help us choose a classifier that will be better (on new, previously unseen, test data)

Regularization and optimization

- **Regularization:**
  - Using a priori knowledge (not just training data) to help us choose a classifier that will be better (on new, previously unseen, test data)
- What type of knowledge?
  - We've seen a very simple "wisdom":
    - "large w are unlikely"
  - We can have more complex forms of knowledge
    - Groups of features, Graphs linking features
- To be able to solve optimization problems that arise with regularization, we need to learn a bit about optimization algorithms

Regularization: bigger picture

- **Regularization:** Useful if training set is limited in size
  - Do we have enough samples to describe all of variability of our two classes in the feature space? **Rarely!**
  - Using **a priori knowledge** (not just training data) helps us choose a classifier that may be better (on new, previously unseen, test data)
- *Regularization (penalty on w)* vs. *No regularization*

  e.g. maximum a posteriori (MAP) vs. e.g. *maximum likelihood (ML)*

  Before seeing the training set (i.e., *a priori*) we know something about "good" **w**'s (or, more generally, good h(x)'s). That is, we prefer some **w**'s / h(x)'s to others, irrespective of training data.

  That knowledge is used together with information contained in the training set to select (*a posteriori*, i.e., after seeing the data) the best **w** (more generally, select the best classifier h(x)).

  All **w**'s (or, more generally, all classifiers h(x)) are *a priori* (i.e. before seeing the training set) equally good. We have no preference.

  Let training set be the only judge of which **w** is best, and which **w** (or more generally, which classifier h(x)) should be chosen.

- For regularization to be beneficial, a prior knowledge should be in correspondence with reality of the classification problem
  - If we assume something that is way off, we're going to make poorer predictions

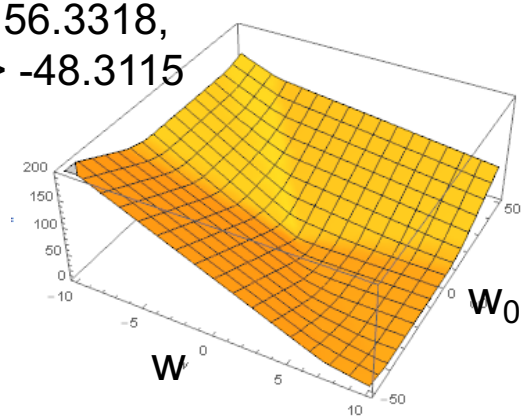
Regularization

- Regularized logistic regression: the algorithm minimizes:

  penalty for complexity of h() (not based on training data) + empirical risk of h() (based on training data)

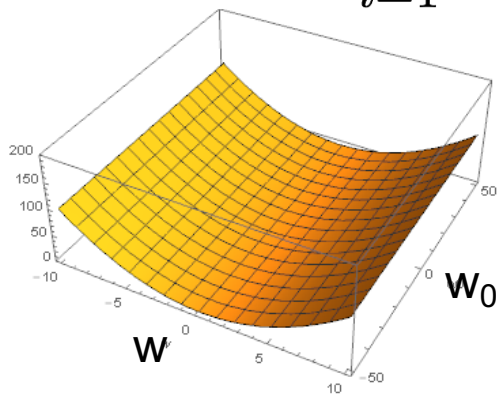
$$\arg\min_w \frac{1}{C}\|w\|_p^p + \sum_{i=1}^{m} \ln(1+\mathrm{e}^{-y_i w^T x_i})$$

Minimum:
w -> 56.3318,
w_0 -> -48.3115



$$\sum_{i=1} \ln(1+\mathrm{e}^{-y_i w^T x_i})$$

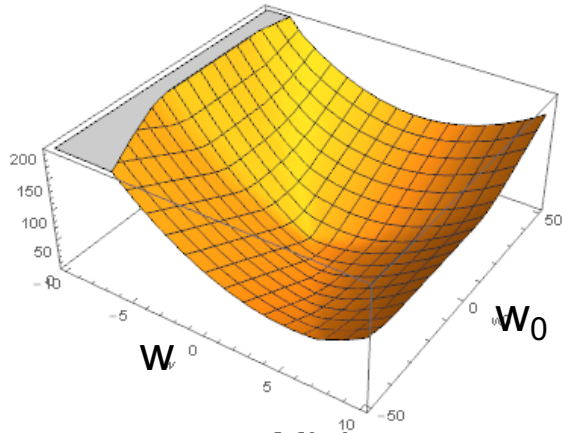
+



$$\|w\|_2^2$$

No penalty over w_0, just over w

=



Minimum:
w -> 0.647365,
w_0 -> -1.14344

- Here, our a priori assumptions is that in the real world that generated the classification problem, **w** is small, so we have a penalty that increases with the norm of **w**

Regularization formulations

- Tikhonov regularization (most popular):

$$\min_w \frac{1}{C}\|w\|_p^p + \sum_{i=1}^{m} \ln(1 + \mathrm{e}^{-y_i w^T x_i})$$

- Ivanov regularization:

$$\min_w \sum_{i=1}^{m} \ln(1 + \mathrm{e}^{-y_i w^T x_i})$$
$$\text{subject to } \|w\|_p^p \le \tau$$

- Morozov regularization:

$$\min_w \|w\|_p^p$$
$$\text{subject to } \sum_{i=1}^{m} \ln(1 + \mathrm{e}^{-y_i w^T x_i}) \le \delta$$

Recap: Regularization

- Let's assume: $P(y_i \,|x_i, w\,) = \dfrac{1}{1+\mathrm{e}^{-y_i w^T x_i}}$
- *Maximum a posteriori (MAP)* estimate of **w:**

$$\arg\min_w \left[\ln \frac{1}{P(w)}\right] + \sum_{i=1}^{m} \ln(1+\mathrm{e}^{-y_i w^T x_i})$$

$$P(w) = \frac{1}{C}\mathrm{e}^{-\frac{1}{C}\|w\|_p^p}$$
$$\ln \frac{1}{P(w)} = \frac{1}{C}\|w\|_p^p + \ln C$$

- Logistic regression

$$\arg\min_w \frac{1}{C}\|w\|_p^p + \sum_{i=1}^{m} \ln(1+\mathrm{e}^{-y_i w^T x_i})$$

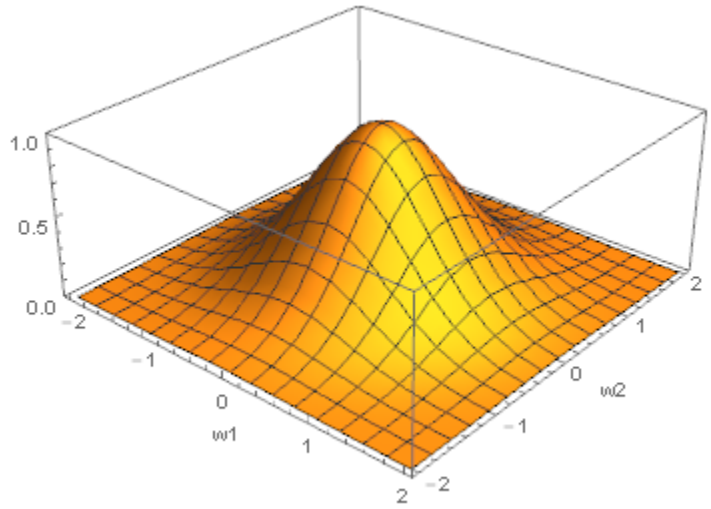
with **regularization** using L_p norm of w

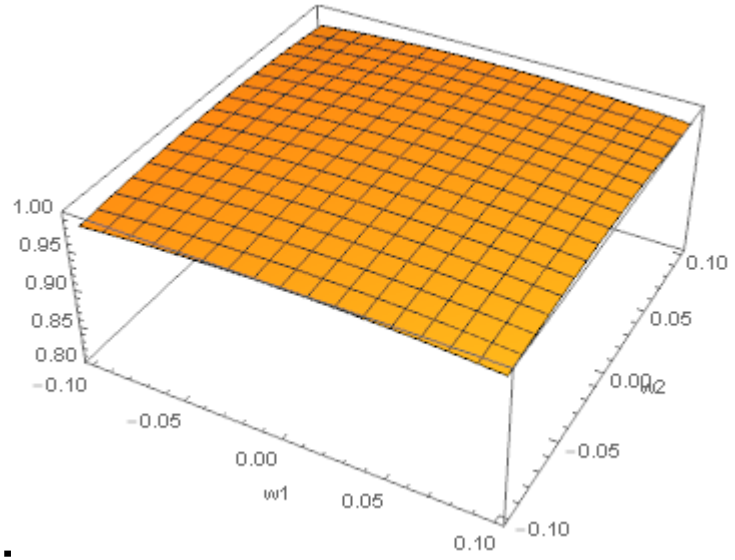
$$P(w) = \frac{1}{C}\mathrm{e}^{-\frac{1}{C}\|w\|_p^p} \qquad \|w\|_p^p = \left(\sum_{f=1}^{F} |w_f|^p\right)$$

L_1 vs L_2 regularization

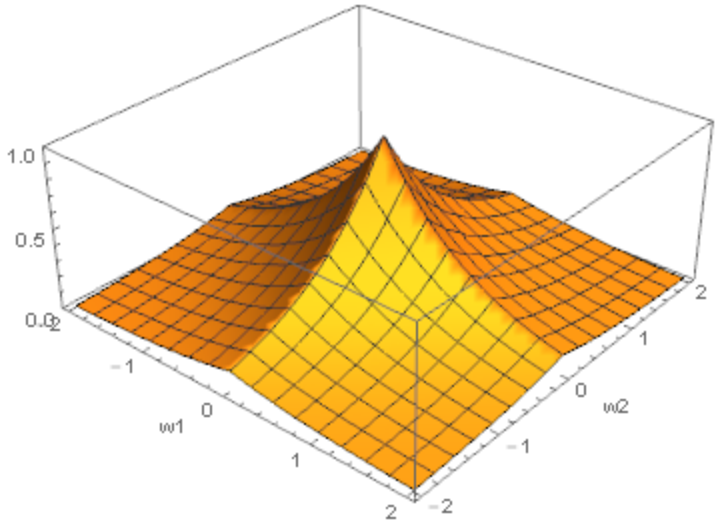
$$P(w) = \frac{1}{C} e^{-\frac{1}{C}\|w\|_p^p}$$

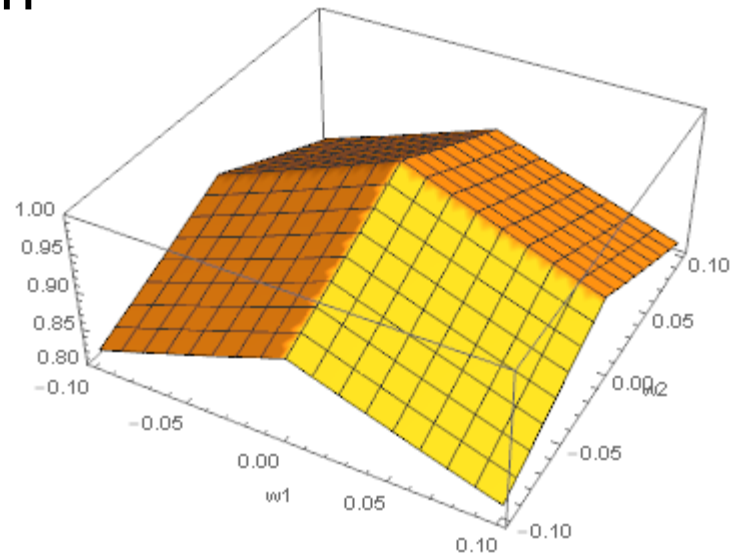
- P(w) for L_2 regularization





- P(w) for L_1 regularization





L_1 vs L_2

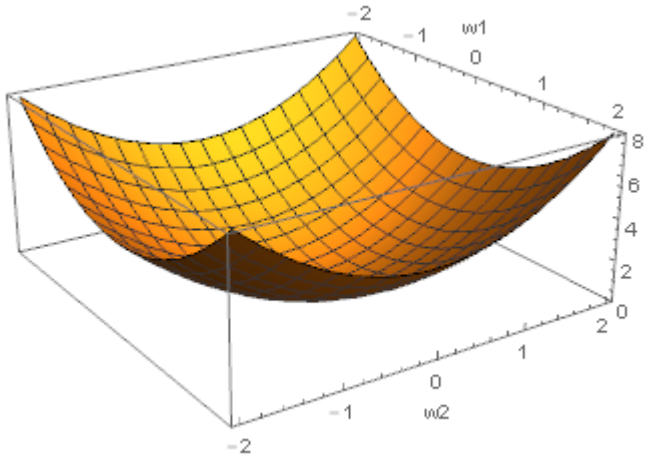
$$\ln \frac{1}{P(w)} = \frac{1}{C}\|w\|_p^p + \ln C$$

- Logistic regression with L_p regularization

$$\arg\min_w \left[\ln \frac{1}{P(w)}\right] + \sum_{i=1}^{m} \ln(1 + \mathrm{e}^{-y_i w^T x_i})$$

$$\arg\min_w \frac{1}{C}\|w\|_p^p + \sum_{i=1}^{m} \ln(1 + \mathrm{e}^{-y_i w^T x_i})$$

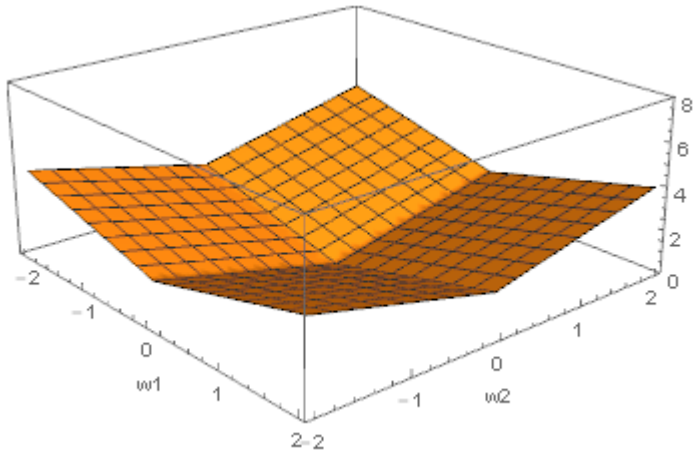
- We're adding a penalty term $\frac{1}{C}\|w\|_p^p$
- The penalty term looks like this:



$$\|w\|_2^2$$

$$\|w\|_p^p = \left(\sum_{f=1}^{F} |w_f|^p\right)$$

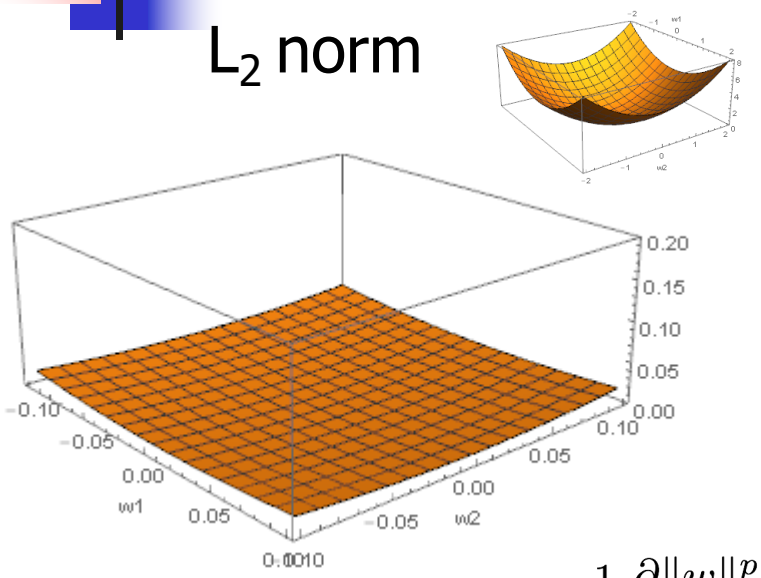
or this:



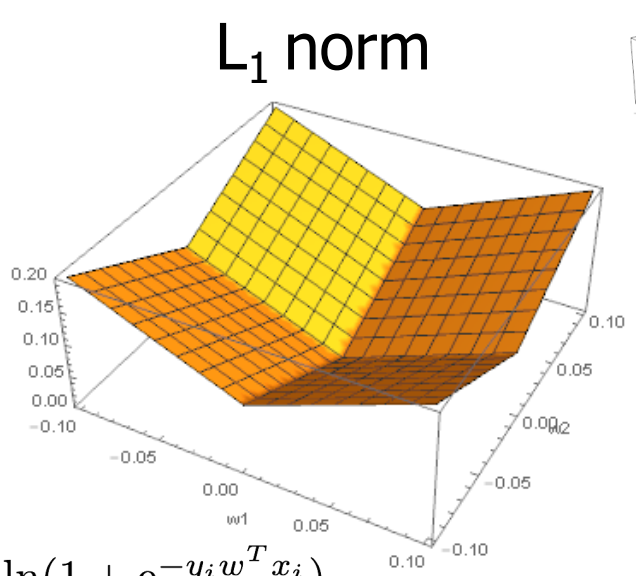
$$\|w\|_1$$

L_1 vs L_2 regularization

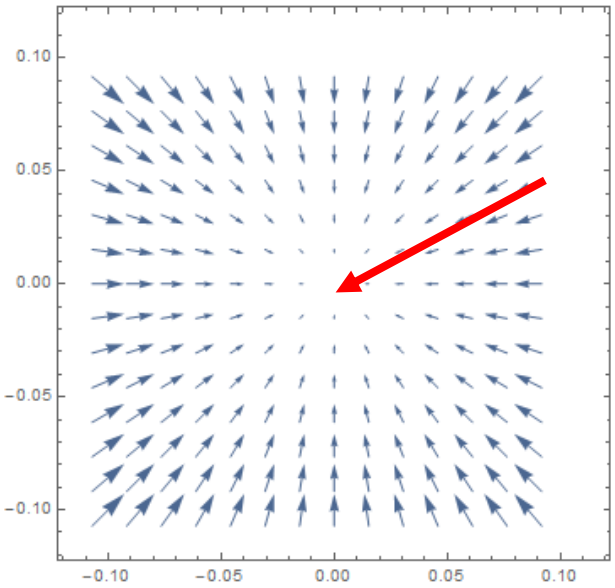
L_2 norm



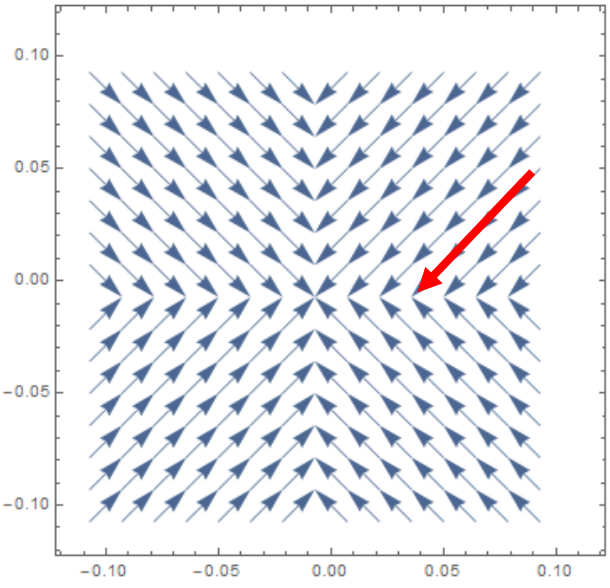
L_1 norm



$$w_{t+1} = w_t - \frac{1}{C}\frac{\partial \|w\|_p^p}{\partial w} + \frac{\partial \sum_{i=1}^{m} \ln(1 + \mathrm{e}^{-y_i w^T x_i})}{\partial w}$$



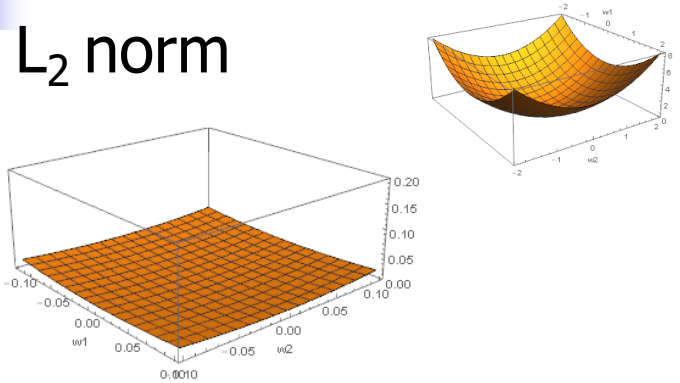
$$-\frac{\partial \|w\|_2^2}{\partial w}$$



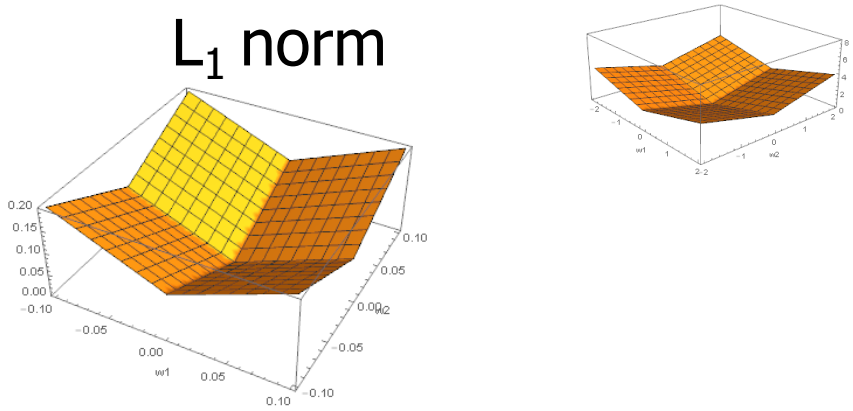
$$-\frac{\partial \|w\|_1}{\partial w}$$

L_1 vs L_2 regularization

L_2 norm

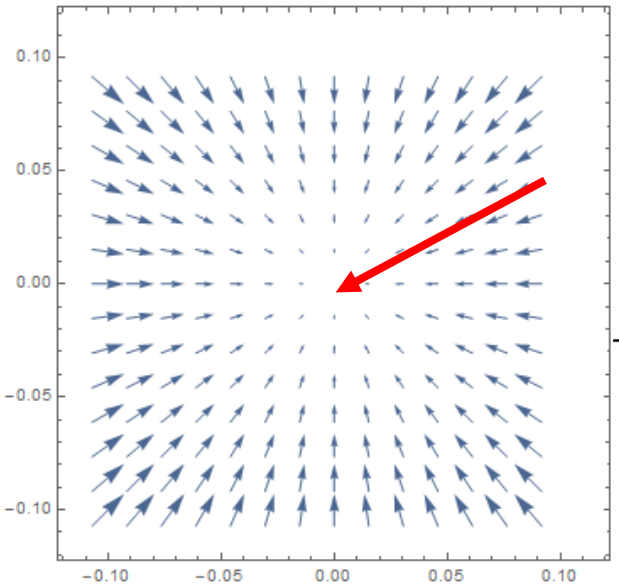


L_1 norm



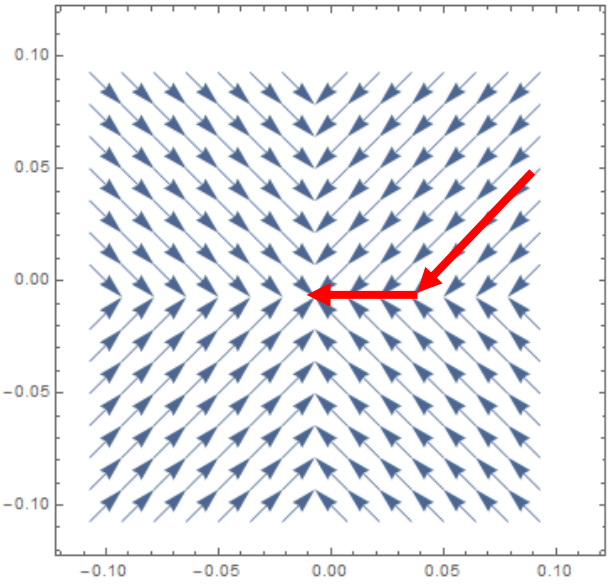
$$w_{t+1} = w_t - \frac{1}{C}\frac{\partial \|w\|_p^p}{\partial w} + \frac{\partial \sum_{i=1}^{m} \ln(1 + e^{-y_i w^T x_i})}{\partial w}$$

All weights small, but not 0



$$- \frac{\partial \|w\|_2^2}{\partial w}$$

some weights not that small, some at 0

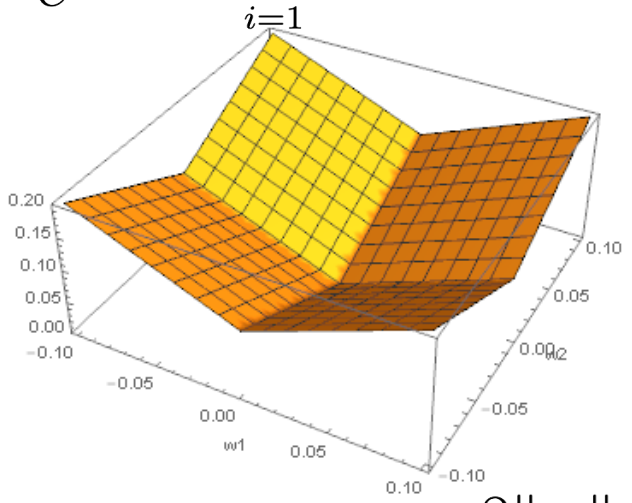


$$- \frac{\partial \|w\|_1}{\partial w}$$

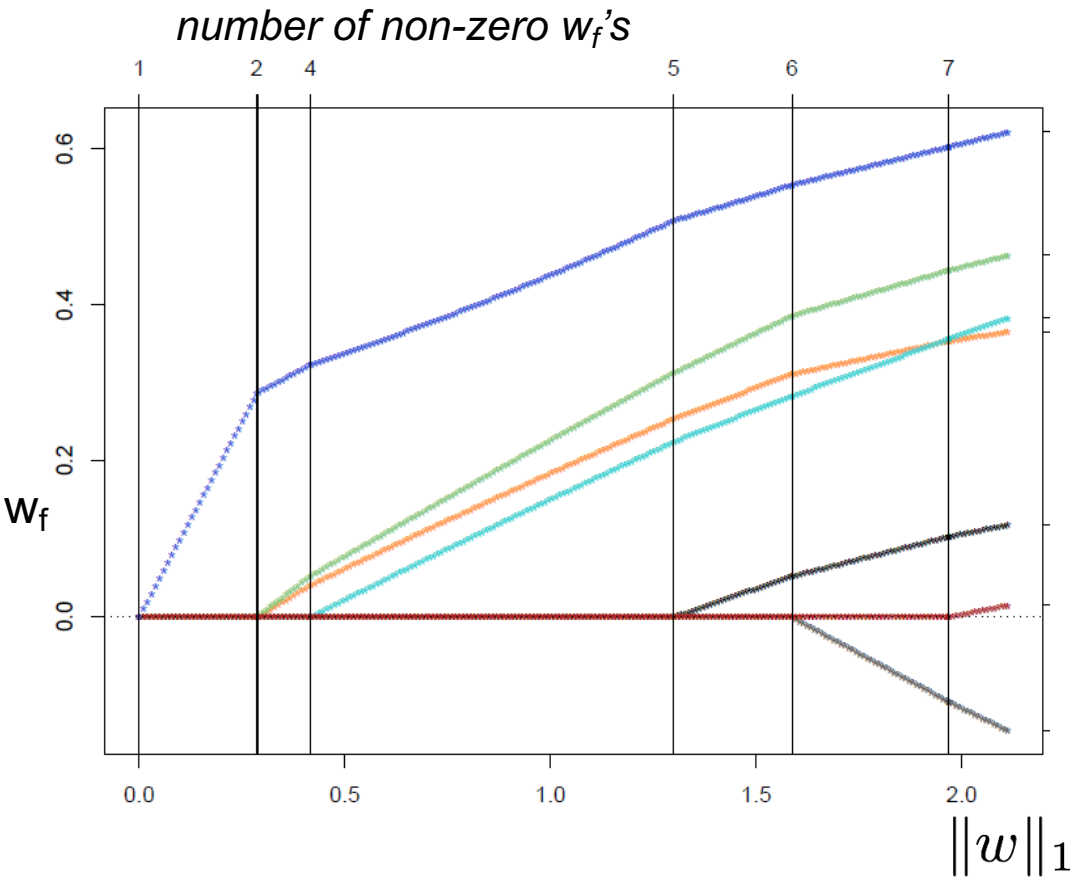
L_1 regularization: feature selection

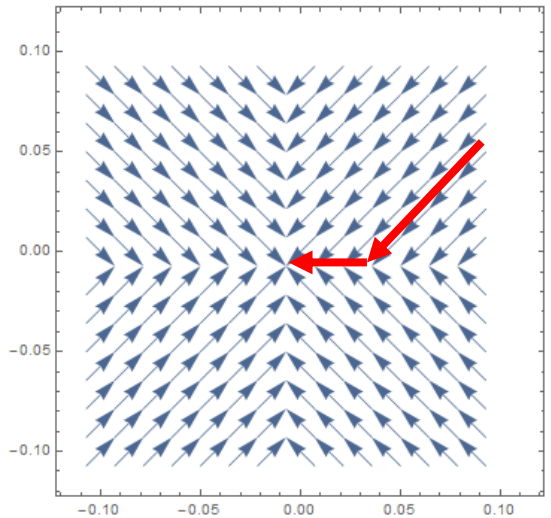
- Using L_1 loss with low value of C leads to some w_f becoming 0
  - feature is no longer used if w_f=0

$$\arg\min_w \frac{1}{C}\|w\|_p^p + \sum_{i=1}^{m} \ln(1 + e^{-y_i w^T x_i})$$



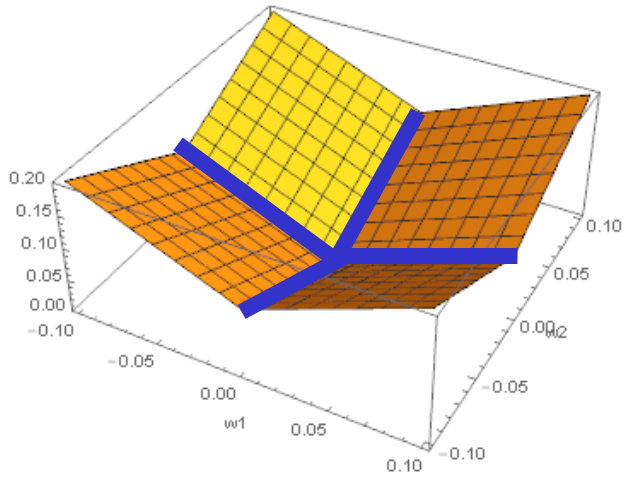
$$-\frac{\partial\|w\|_1}{\partial w}$$

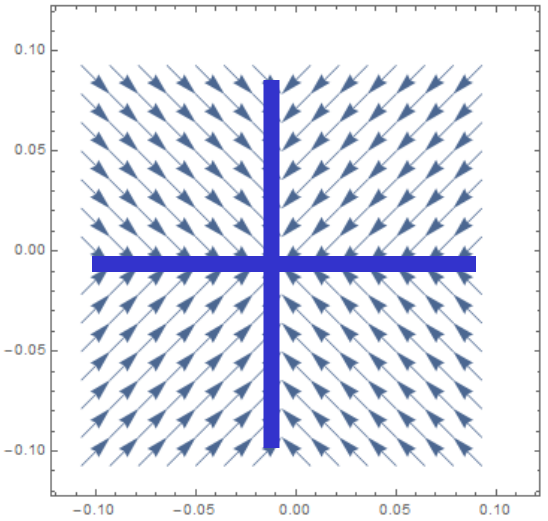




Non-smooth optimization

- If a convex function is smooth (here: is differentiable), we can calculate gradient at any point, and use it to go towards global minimum
- For non-smooth functions, there will be points for which we can't calculate gradient (some partial derivatives don't exist)

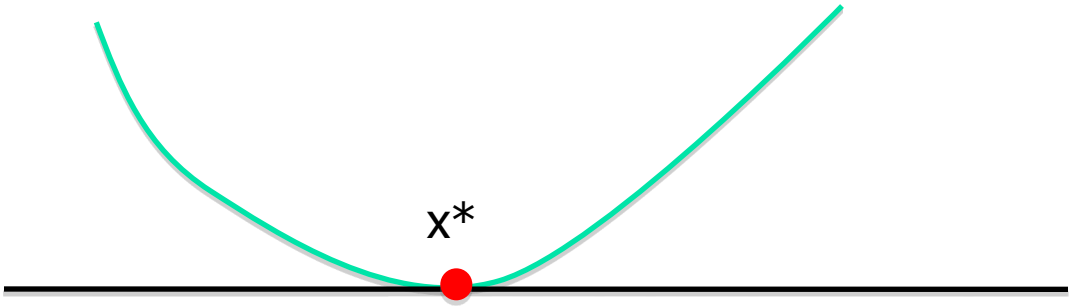




Optimization refresher

Necessary condition for global optimum in unconstrained optimization:

If x^* is a global optimum, then $\nabla f(x^*) = 0$



Optimization is much easier if the necessary condition for global optimum is also sufficient

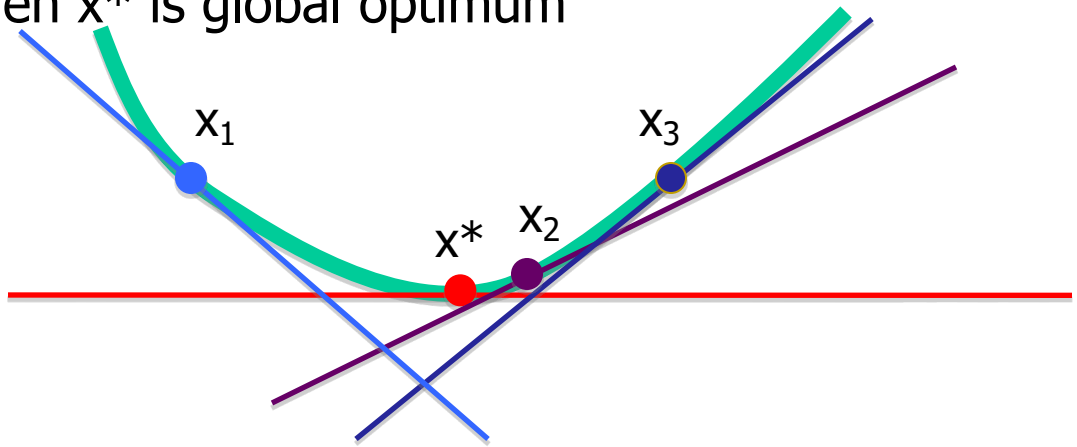
if $\nabla f(x^*) = 0$, then x^* is global optimum

Are there classes of functions with the above?

Convex optimization

For all **convex** functions f:

if $\nabla f(x^*) = 0$, then x^* is global optimum



f is **convex** if all its values f(y) are (weakly) above any gradient-based linear approximation of f

$f(y) >= f(x^*) + \langle \nabla f(x^*), y-x^* \rangle$ for all y

$f(y) >= f(x_1) + \langle \nabla f(x_1), y-x_1 \rangle$ for all y

$f(y) >= f(x_2) + \langle \nabla f(x_2), y-x_2 \rangle$ for all y

$f(y) >= f(x_3) + \langle \nabla f(x_3), y-x_3 \rangle$ for all y

For all x,y:

$f(y) >= f(x) + \langle \nabla f(x), y-x \rangle$

Convexity (skip)

<x,z+y>=<x,z>+<x,y>
<ax,by>=ab<x,y> for real a,b;
<x,y>=<y,x>

f is convex iff: $f(y) \geq f(x) + \langle \nabla f(x), y-x \rangle$ for all x,y

That means:

$f(\beta x+(1-\beta)y) \leq \beta f(x) + (1-\beta) f(y)$ for any β in [0,1]

Proof: Let $z = \beta x+(1-\beta)y = \beta x+y-\beta y$

$f(z) \leq f(y) - \langle \nabla f(z), y-z \rangle \quad \Leftrightarrow \quad f(y) \geq f(z) + \langle \nabla f(z), y-z \rangle$

$f(z) \leq f(y) - \langle \nabla f(z), y-(\beta x+y-\beta y) \rangle = f(y) - \langle \nabla f(z), \beta y-\beta x) \rangle$

$f(z) \leq f(y) - \beta \langle \nabla f(z), y-x \rangle$

$(1-\beta) f(z) \leq (1-\beta) f(y) - (1-\beta) \beta \langle \nabla f(z), y-x \rangle$ we can multiply because β in [0,1]

$f(z) \leq f(x) - \langle \nabla f(z), x-z \rangle \quad \Leftrightarrow \quad f(x) \geq f(z) + \langle \nabla f(z), x-z \rangle$

$f(z) \leq f(x) - \langle \nabla f(z), x-\beta x-(1-\beta)y) \rangle = f(x) - \langle \nabla f(z), (1-\beta)(x-y)) \rangle$

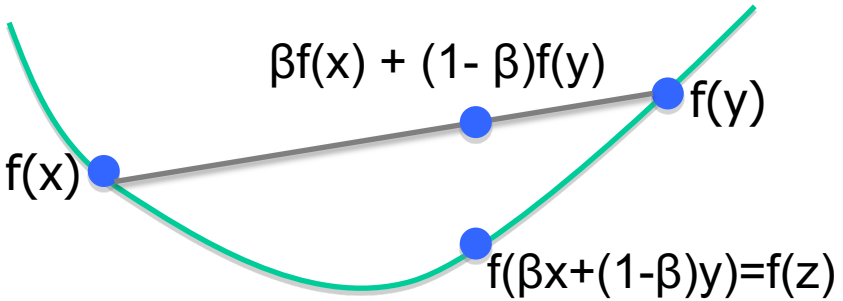
$f(z) \leq f(x) + (1-\beta) \langle \nabla f(z), y-x \rangle$

$\beta f(z) \leq \beta f(x) + (1-\beta) \beta \langle \nabla f(z), y-x \rangle$

Add green and blue

$f(z) \leq \beta f(x) + (1-\beta) f(y)$

$f(\beta x+(1-\beta)y) \leq \beta f(x) + (1-\beta) f(y)$



Convexity

f is convex iff: $f(y) \geq f(x) + \langle \nabla f(x), y-x \rangle$ for all x,y

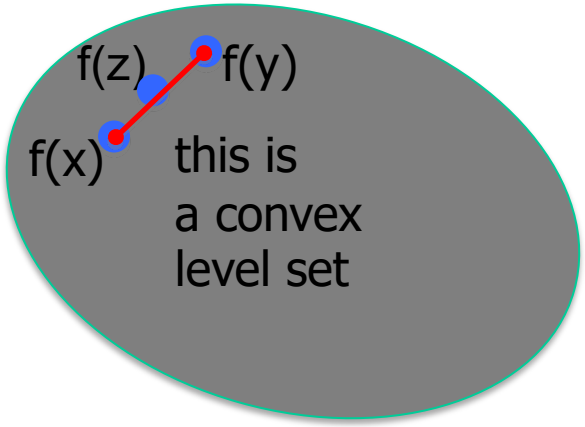
That means:

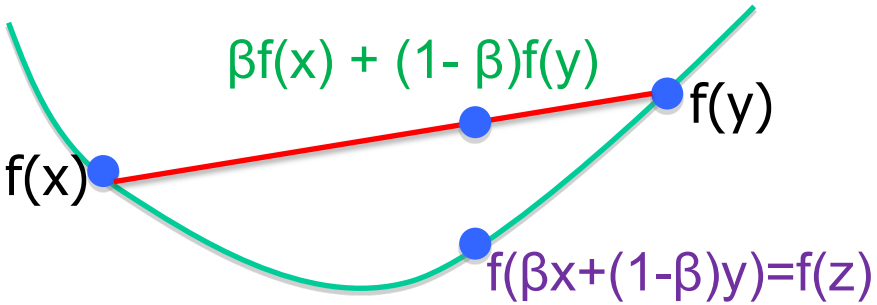
$f(\beta x+(1-\beta)y) \leq \beta f(x) + (1-\beta) f(y)$ for any β in [0,1]

Let level set $L_f(\alpha)=\{x: f(x)<= \alpha\}$

$L_f(\alpha)$ is part of the function domain where function has value $\leq \alpha$

If f is convex, level sets are convex





Convexity

f is convex iff: $f(y) \geq f(x) + \langle \nabla f(x), y-x \rangle$ for all x,y

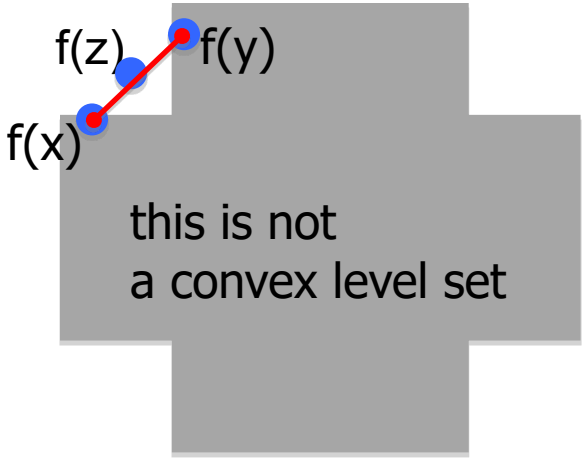
That means:

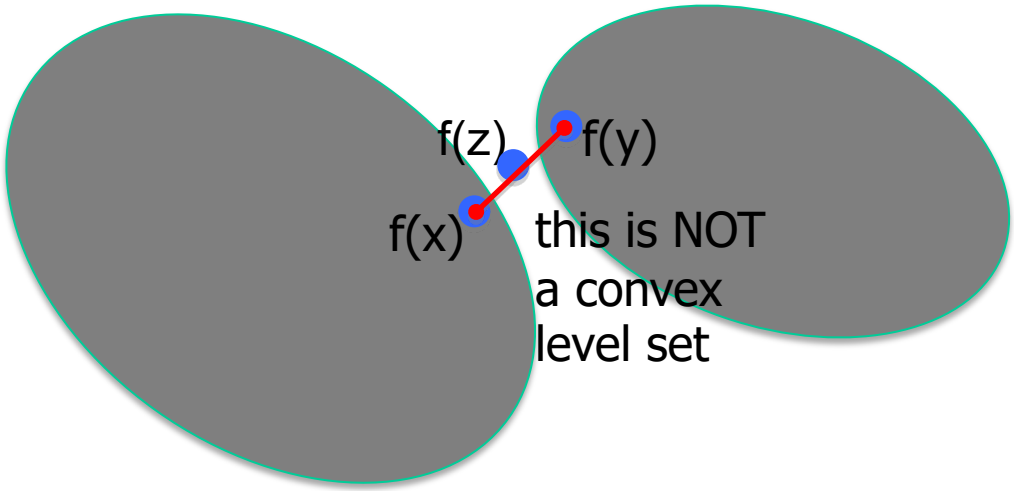
$f(\beta x+(1-\beta)y) \leq \beta f(x) + (1-\beta)f(y)$ for any β in [0,1]

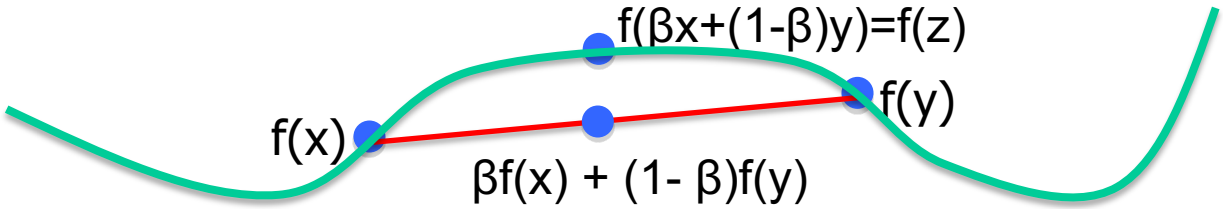
Let **level set** $L_f(\alpha)=\{x: f(x)<= \alpha\}$

$L_f(\alpha)$ is part of the function domain where function has value $\leq \alpha$

If f is convex, level sets are convex





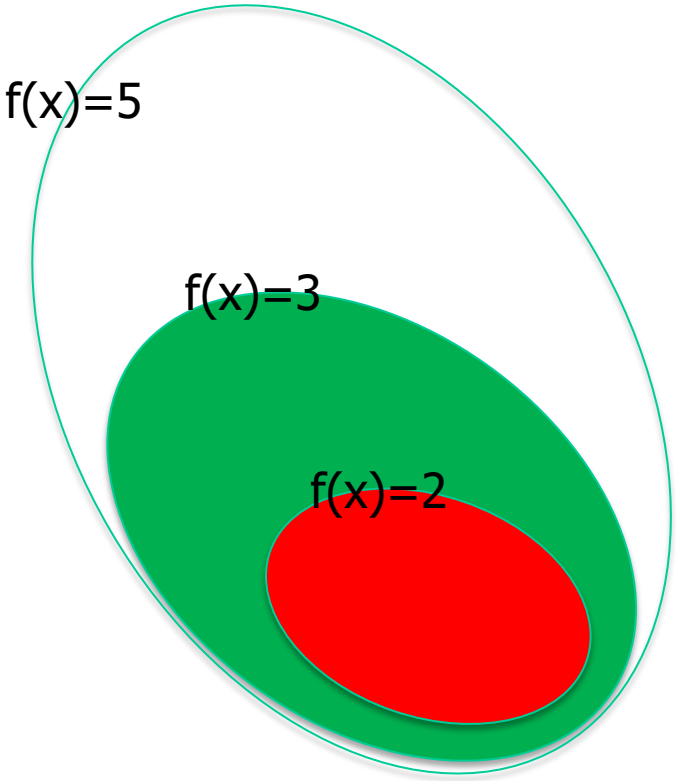


Convex optimization

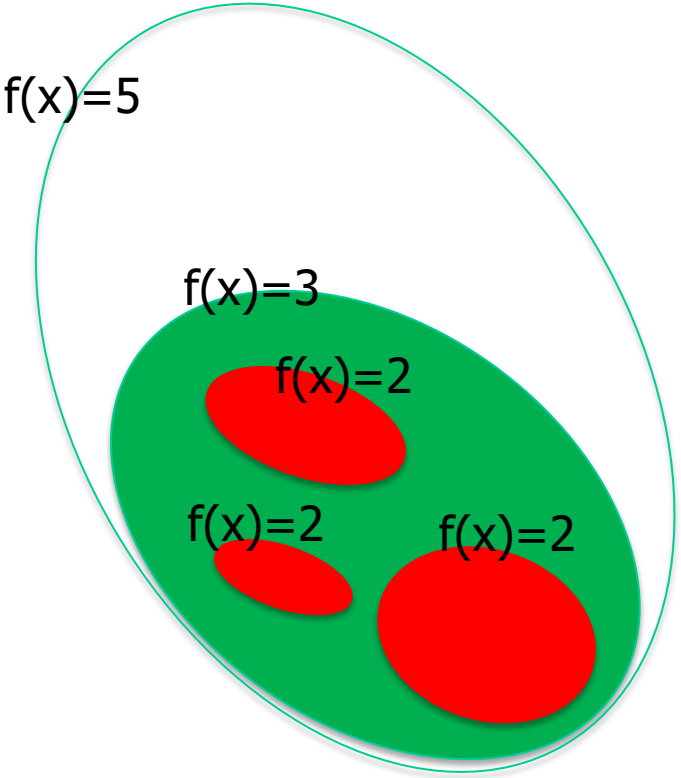
Let level set $L_f(\alpha)=\{x: f(x)<= \alpha\}$

If $\alpha<\beta$, then $L_f(\alpha)$ is a subset of $L_f(\beta)$, **both are convex**

We have this situation:



Never this:

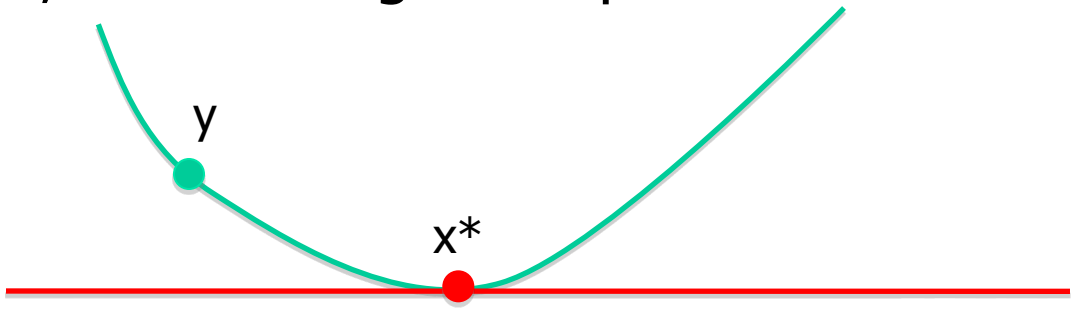


As we go lower, **we never have to backtrack**!

Convex optimization

For all convex functions f:

if $\nabla f(x^*) = 0$, then x^* is global optimum



f is convex if all its values f(y) are above any gradient-based linear approximation of f

Let x^* be a point where $\nabla f(x^*) = 0$, then for all y

$f(y) >= f(x^*)+\langle \nabla f(x^*), y-x^* \rangle = f(x^*) + \langle 0, y-x^* \rangle = f(x^*)$

that is

$f(y) >= f(x^*)$, for all y

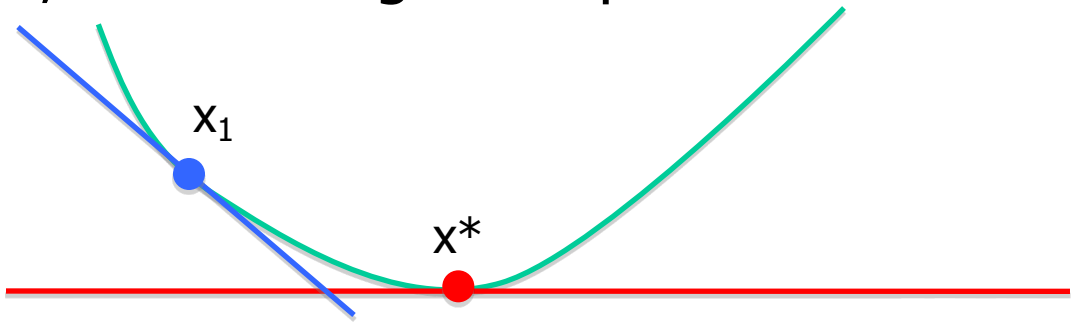
which means x^* is global minimum

$\langle x,0 \rangle = 0$

Convex optimization

For all convex functions f:

if $\nabla f(x^*) = 0$, then x^* is global optimum

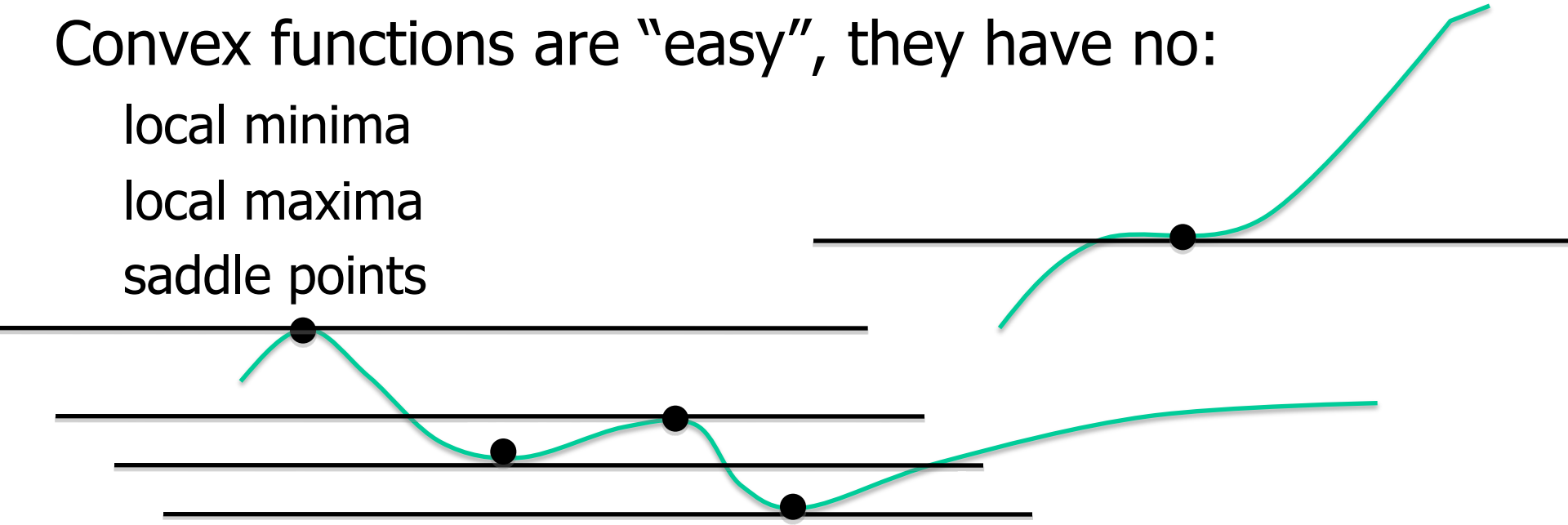


Convex functions are "easy", they have no:

local minima

local maxima

saddle points



What is gradient?

gradient of a n-dimensional function $f:R^n \rightarrow R$ is:

an n-dimensional **vector**

$$\nabla f(\mathbf{z}) = \left(\frac{\partial}{\partial x_1} f(\mathbf{x})|_{\mathbf{x}=\mathbf{z}}, \ldots, \frac{\partial}{\partial x_n} f(\mathbf{x})|_{\mathbf{x}=\mathbf{z}} \right)$$

pointing (for a point x) in direction of steepest slope of f (pointing up)

Defines a line, plane or hyperplane $p(x')=\langle \nabla f(x), x'-x \rangle$ tangent to f at x

If f(x) is convex, the tangent plane is always below f:

$$\forall x' \in \mathbb{R}^n f(x') \geq f(x) + \langle \nabla f(x), x' - x \rangle$$

<x,y> represents inner product (dot product) of two vectors x, y

In 1D, it reduces to simple multiplication

Dot product: coordinate-wise multiplication

Orthogonal vectors x,y have <x,y>=0

Dot product rules:

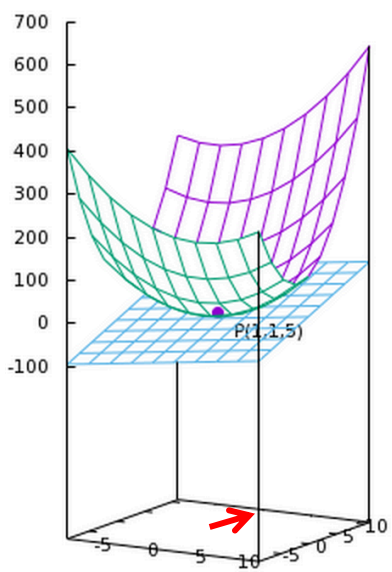
<x,z+y>=<x,z>+<x,y>

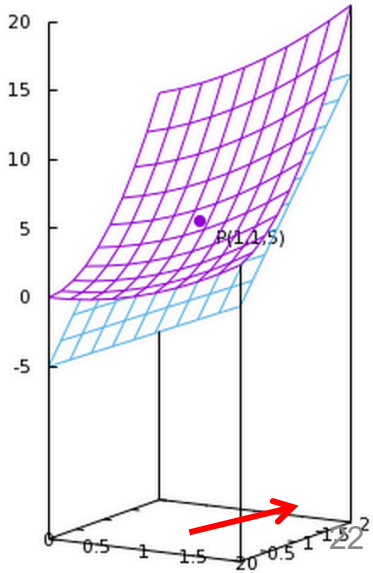
<ax,by>=ab<x,y> for real a,b;

in particular <x,0>=0

$\langle x,x \rangle = ||x||^2$

$\langle x,y \rangle = \langle y,x \rangle = y^T x = x^T y$



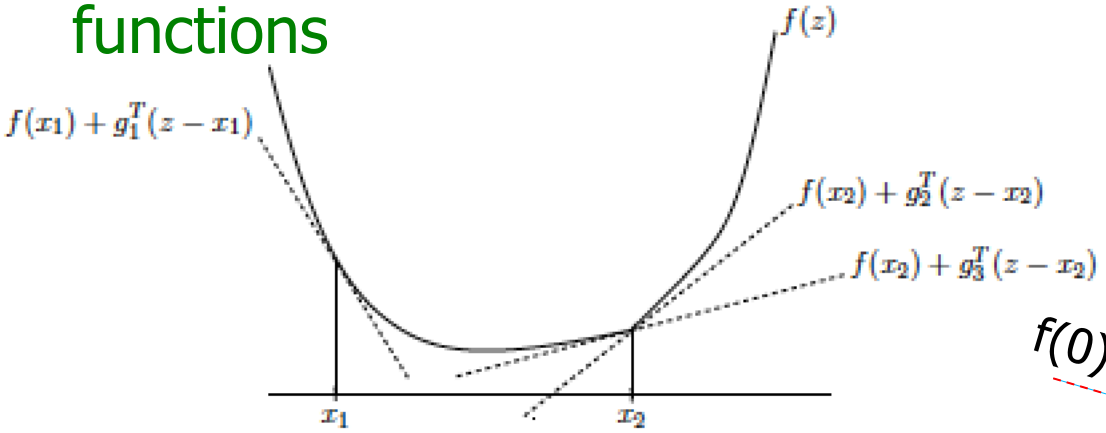


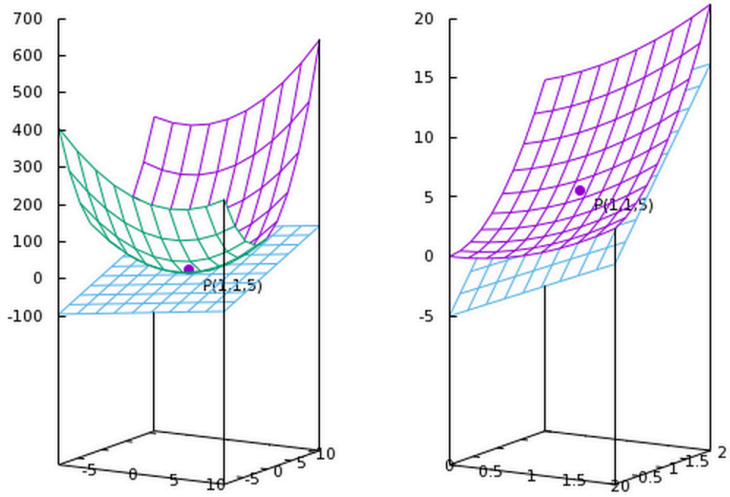
Subgradient

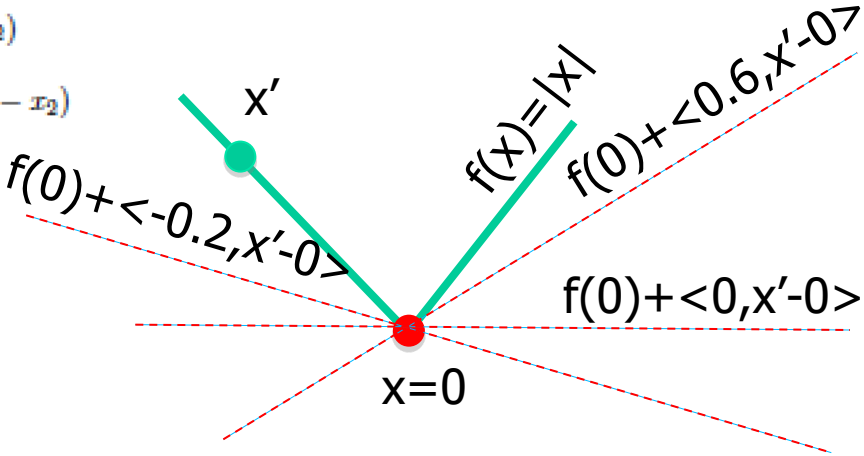
If f(x) is convex, the tangent plane is always below f:

$$\forall x' \in \mathbb{R}^n f(x') \geq f(x) + \langle \nabla f(x), x' - x \rangle$$

We can use that approach to
define something similar to
gradient, for non-differentiable
but continuous, convex (why?)
functions





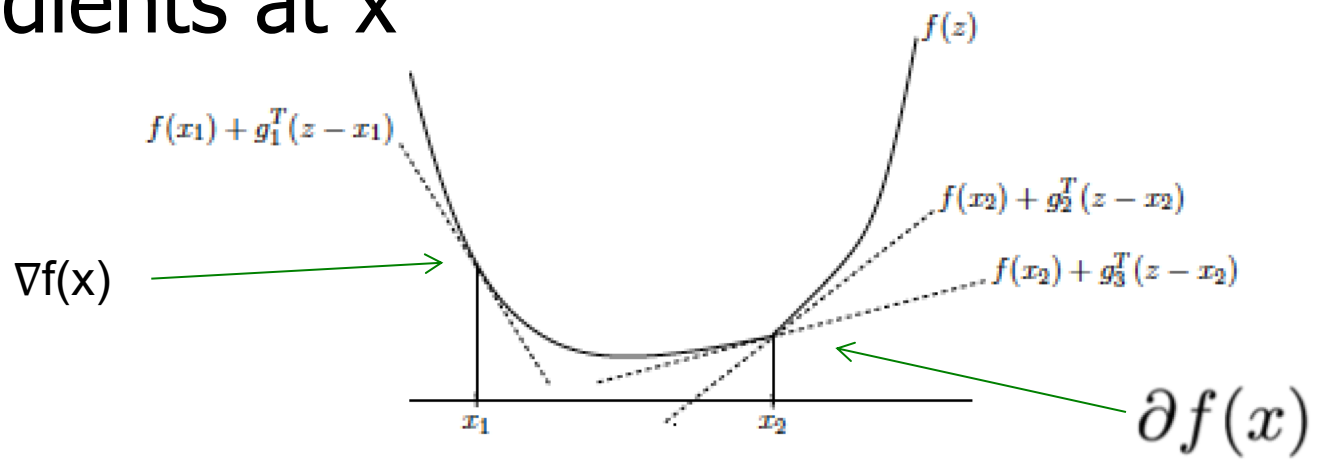


Subgradient of f() at x:
any vector **g** that satisfies:

$$\forall x' \in \mathbb{R}^n \quad f(x') \geq f(x) + \langle g, x' - x \rangle$$

Subdifferential

- A non-differentiable function f() may have many subgradients at x



- **Set of subgradients** at x is called a **subdifferential** of f() at x

$$\partial f(x) = \{g : \forall x' \in \mathbb{R}^n \quad f(x') \geq f(x) + \langle g, x' - x \rangle\}$$

- if gradient ∇f(x) exists at x, it is the only subgradient:
  - Then, subdifferential is a set with only one element

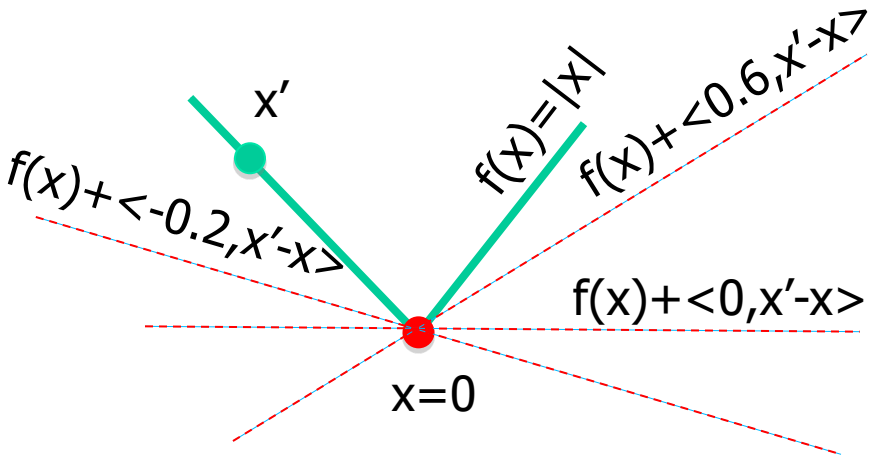
$$\partial f(x) = \{\nabla f(x)\}$$

Subdifferential

- Simple example (1D): for f(x)=|x|, what is g?

$$f(x) = |x|$$
$$f(x') \geq f(x) + \langle g, x' - x \rangle$$
$$|x'| \geq 0 + gx'$$
$$|x'| \geq gx'$$
$$1 \geq g \, \mathrm{sign}(x')$$
$$1 \geq g$$
$$-1 \leq g$$
$$1 \geq |g|$$
$$\partial f(0) = \{g : |g| \leq 1\}$$

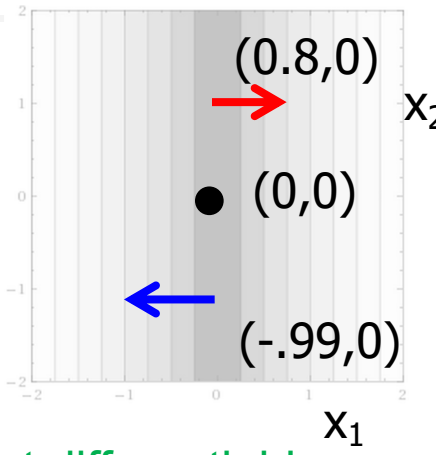
- 1D, so subgradient g is just a real number
  - Subdifferential is a set of numbers



Subdifferential

- Simple example (2D):

$$f_1(x = (x_1, x_2)) = |x_1| \quad g = (g_1, g_2)$$
$$f(x') \geq f(x) + \langle g, x' - x \rangle$$
$$|x_1'| \geq |x_1| + g_1(x_1' - x_1) + g_2(x_2' - x_2)$$
$$x_1 = 0$$

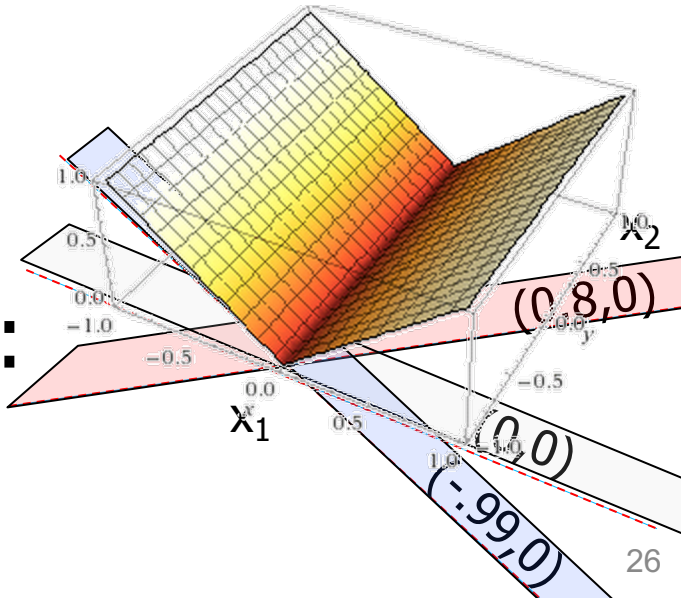


that's the interesting point where f is not differentiable

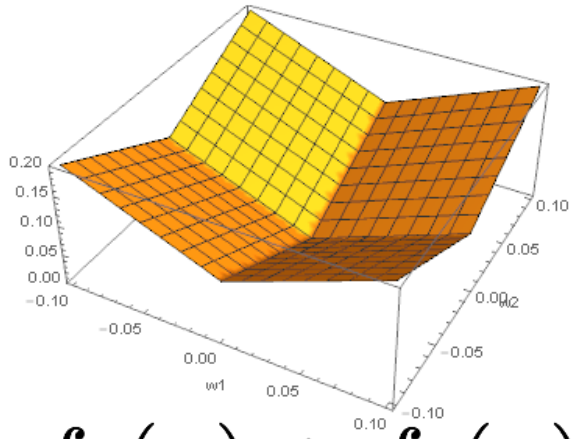
$$|x_1'| \geq g_1 x_1' + g_2(x_2' - x_2)$$
$$(x_2' - x_2) \quad \text{unbounded} \implies g_2 = 0$$
$$|x_1'| \geq g_1 x_1'$$
$$|g_1| \leq 1$$
$$\partial f_1((0, x_2)) = \{(g, 0) : |g| \leq 1\}$$

- Similar for the other variable:

$$f_2(x) = |x_2|$$
$$\partial f_2((x_1, 0)) = \{(0, g) : |g| \leq 1\}$$



Subdifferential



- L_1 norm: $f_1(x) = |x_1|$

$$f_2(x) = |x_2|$$

$$f(x) = |x_1| + |x_2| = f_1(x) + f_2(x)$$

- We need to know how subdifferential behaves for sum of functions
  - if: $f(x) = f_1(x) + f_2(x)$
  - then: $\partial f(x) = \partial f_1(x) + \partial f_2(x)$

algebraic (Minkowski) sum of sets :

$$A + B = \{a + b : a \in A, b \in B\} \qquad A + \emptyset = \emptyset$$

- Easy to check from definition:

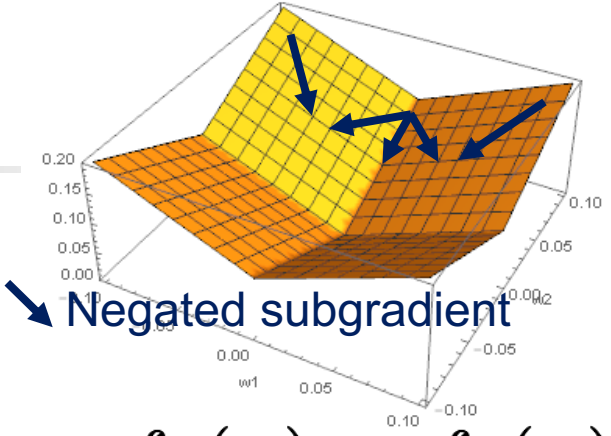
using: $\langle g_1, z\rangle + \langle g_2, z\rangle = \langle g_1 + g_2, z\rangle$

$$\partial f(x) = \{g : \forall x' \in \mathbb{R}^n \quad f(x') \geq f(x) + \langle g, x' - x\rangle\}$$

Subdifferential

- L_1 norm:

$$f_1(x) = |x_1|$$
$$f_2(x) = |x_2|$$
$$f(x) = |x_1| + |x_2| = f_1(x) + f_2(x)$$



- Here's what we derived for f_1 and f_2:

$$\partial f_1((0, x_2)) = \{(g, 0) : |g| \leq 1\}$$
$$\partial f_1((x_1 \neq 0, x_2)) = \{\nabla f_1(x)\} = \{(\text{sign}(x_1), 0)\} = \{(g, 0) : g = \text{sign}(x_1)\}$$
$$\partial f_2((x_1, 0)) = \{(0, g) : |g| \leq 1\}$$
$$\partial f_2((x_1, x_2 \neq 0)) = \{\nabla f_2(x)\} = \{(0, \text{sign}(x_2))\} = \{(0, g) : g = \text{sign}(x_2)\}$$

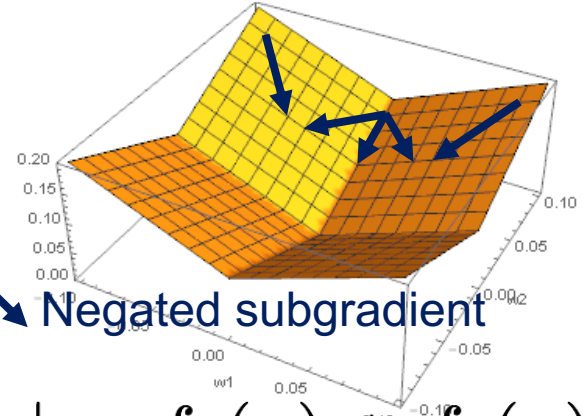
- Together, the final result for f():

$$\partial f((0, 0)) = \{(g_1, g_2) : |g_1| \leq 1, |g_2| \leq 1\}$$
$$\partial f((0, x_2)) = \{(g, \text{sign}(x_2)) : |g| \leq 1\}$$
$$\partial f((x_1, 0)) = \{(\text{sign}(x_1), g) : |g| \leq 1\}$$
$$\partial f((x_1, x_2)) = \{(\text{sign}(x_1), \text{sign}(x_2))\} = \{\nabla f(x)\}$$

Subdifferential

- L_1 norm (2D):

$$f_1(x) = |x_1|$$
$$f_2(x) = |x_2|$$
$$f(x) = |x_1| + |x_2| = f_1(x) + f_2(x)$$



$$\partial f((0,0)) = \{(g_1, g_2) : |g_1| \leq 1, |g_2| \leq 1\}$$
$$\partial f((0, x_2)) = \{(g, \text{sign}(x_2)) : |g| \leq 1\}$$
$$\partial f((x_1, 0)) = \{(\text{sign}(x_1), g) : |g| \leq 1\}$$
$$\partial f((x_1, x_2)) = \{(\text{sign}(x_1), \text{sign}(x_2))\} = \{\nabla f(x)\}$$

- More readable form:

$$\partial f(x) = G_1 \times G_2$$
$$G_i(x) = \begin{cases} [-1, 1], & \text{if } x_i = 0 \\ \{\text{sign}(x_i)\}, & \text{if } x_i \neq 0 \end{cases}$$

- L_1 norm (n-D):

$$f : \mathbb{R}^n \to \mathbb{R}$$
$$f(x) = \sum_{i=1}^{n} |x_i|$$
$$\partial f(x) = G_1(x) \times G_2(x) \times \ldots \times G_n(x)$$

Condition for Global minimum

- Convex f: differentiable risk + (non-) differentiable penalty
- Necessary and sufficient condition for global minimum for continuous, convex f:

$$f(w) = \hat{R}_{S_m}(w) + \Omega(w)$$

$$\partial f(w) = \left\{\nabla_w \hat{R}_{S_m}(w)\right\} + \partial\Omega(w)$$

$$w^* = \arg\min_w f(w) \Leftrightarrow 0 \in \partial f(w^*)$$

$$0 \in \left\{\nabla_w \hat{R}_{S_m}(w^*)\right\} + \partial\Omega(w^*)$$

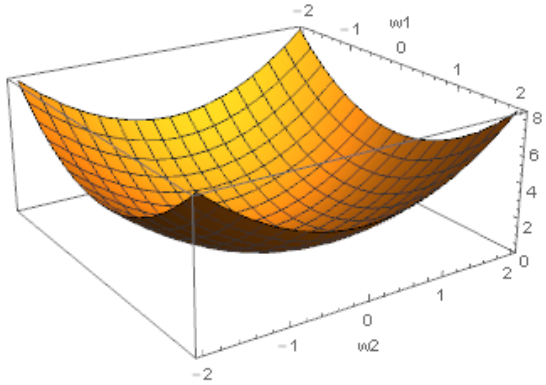
$$-\nabla_w \hat{R}_{S_m}(w^*) \in \partial\Omega(w^*)$$

algebraic (Minkowski) sum of sets :

$A + B = \{a + b : a \in A, b \in B\}$ $\quad A + \emptyset = \emptyset$

Condition for Global minimum

- L_2 norm (differentiable)



$$\Omega(w) = ||w||_2^2 = \sum_{i=1}^{n} w_i^2$$

$$-\nabla_w \hat{R}_{S_m}(w^*) = \nabla_w \Omega(w^*)$$

$$-\frac{1}{2}\nabla_w \hat{R}_{S_m}(w^*) = w^*$$

$$-\frac{1}{2}\frac{\partial \hat{R}_{S_m}(w^*)}{\partial w_i} = w_i^*$$

$$-\frac{1}{2}\sum_{j=1}^{m}\frac{\partial \log(1+\exp(-y_j {w^*}^T x_j))}{\partial w_i} = w_i^*$$

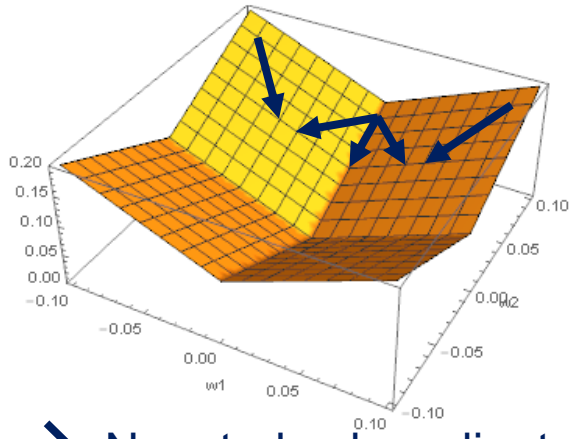
$$w_i^* = 0 \text{ is part of minimum } w^* \text{ if } \frac{\partial \hat{R}_{S_m}(w^*)}{\partial w_i} = 0$$

- Gradient of risk has to exactly cancel gradient of $||w||^2$
  - Unlikely to happen at $w_i=0$
- Built-in feature selection – very unlikely

Subdifferential

- L_1 norm (2D):

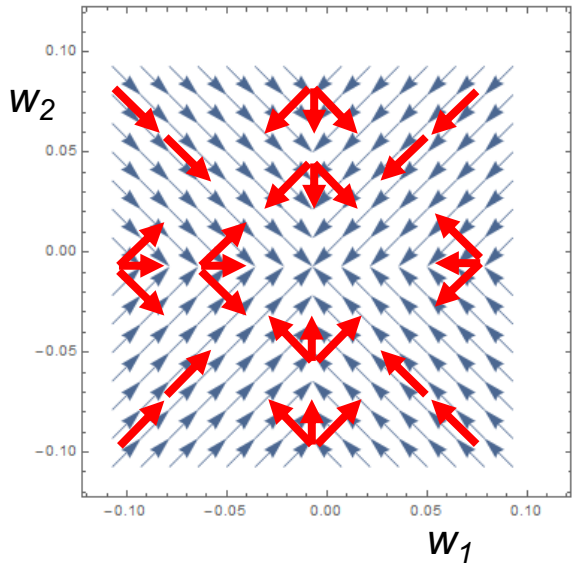
$$f(x) = |x_1| + |x_2|$$



↘ Negated subgradient

$$\partial f(x) = G_1 \times G_2$$

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



↘ Negated subgradient

- L_1 norm (n-D):

$$f : \mathbb{R}^n \rightarrow \mathbb{R}$$

$$f(x) = \sum_{i=1}^{n} |x_i|$$

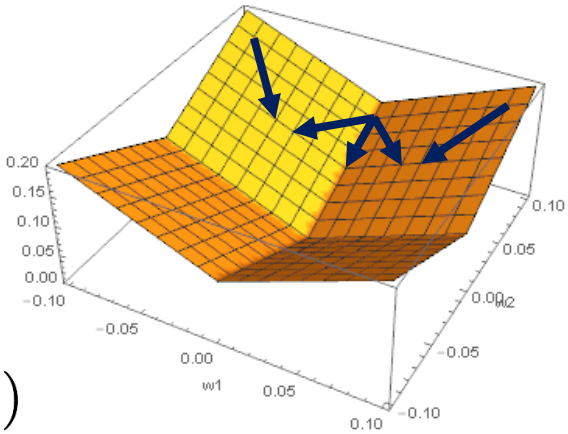
$$\partial f(x) = G_1(x) \times G_2(x) \times \ldots \times G_n(x)$$

Condition for Global minimum

- L_1 penalty (non-differentiable)

$$\Omega(w) = \lambda||w||_1 = \lambda \sum_{i=1}^{n} |w_i|$$

Larger λ = larger penalty for same $||w||_1$



$$-\nabla_w \hat{R}_{S_m}(w^*) \in \partial\Omega(w^*)$$

$$-\nabla_w \hat{R}_{S_m}(w^*) \in \lambda(G_1(w^*) \times G_2(w^*) \times \ldots \times G_n(w^*))$$

$$-\frac{\partial \hat{R}_{S_m}(w^*)}{\partial w_i} \in \lambda G_i(w^*)$$

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

$$\frac{\partial \hat{R}_{S_m}(w^*)}{\partial w_i} \in \begin{cases} [-\lambda,\lambda], & \text{if } w_i^* = 0 \\ \{-\lambda \text{sign}(w_i^*)\}, & \text{if } w_i^* \neq 0 \end{cases}$$

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

- Feature selection – no longer so unlikely!
  - Gradient of risk has a range to fall into
  - Larger λ=>wider range λG_i => $w_i^*=0$ more likely

Condition for Global minimum

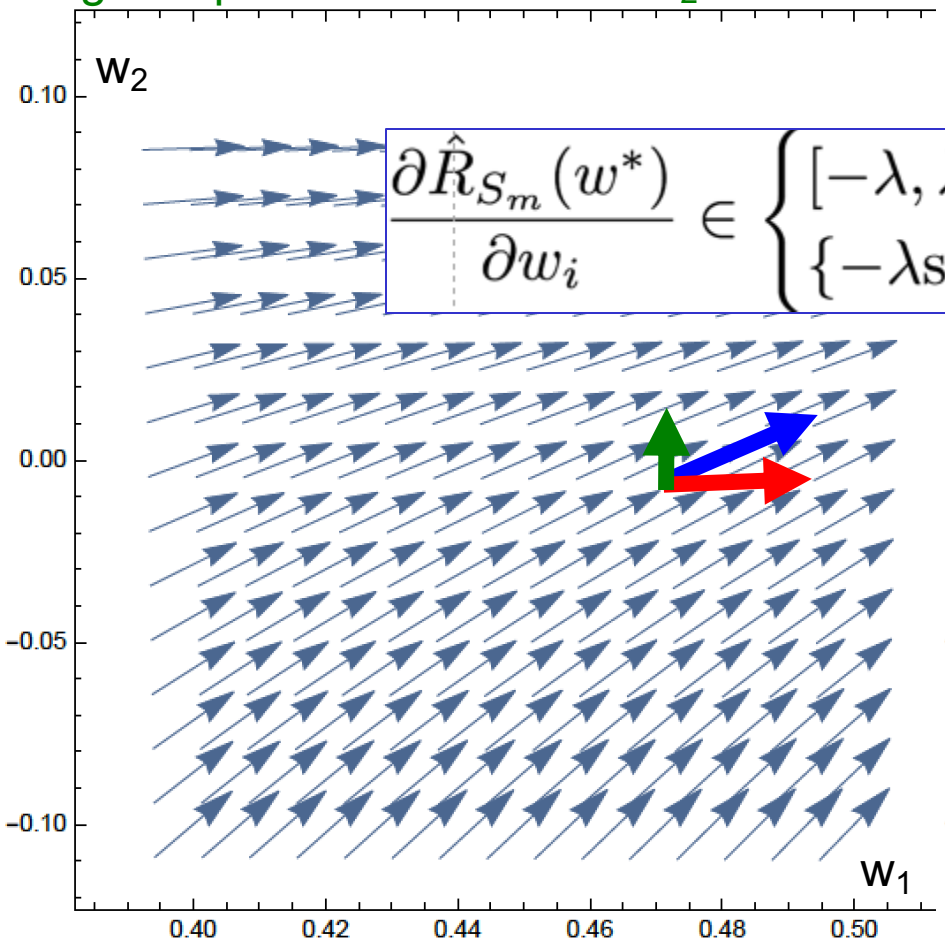
Risk + L_1 penalty: gradient plots

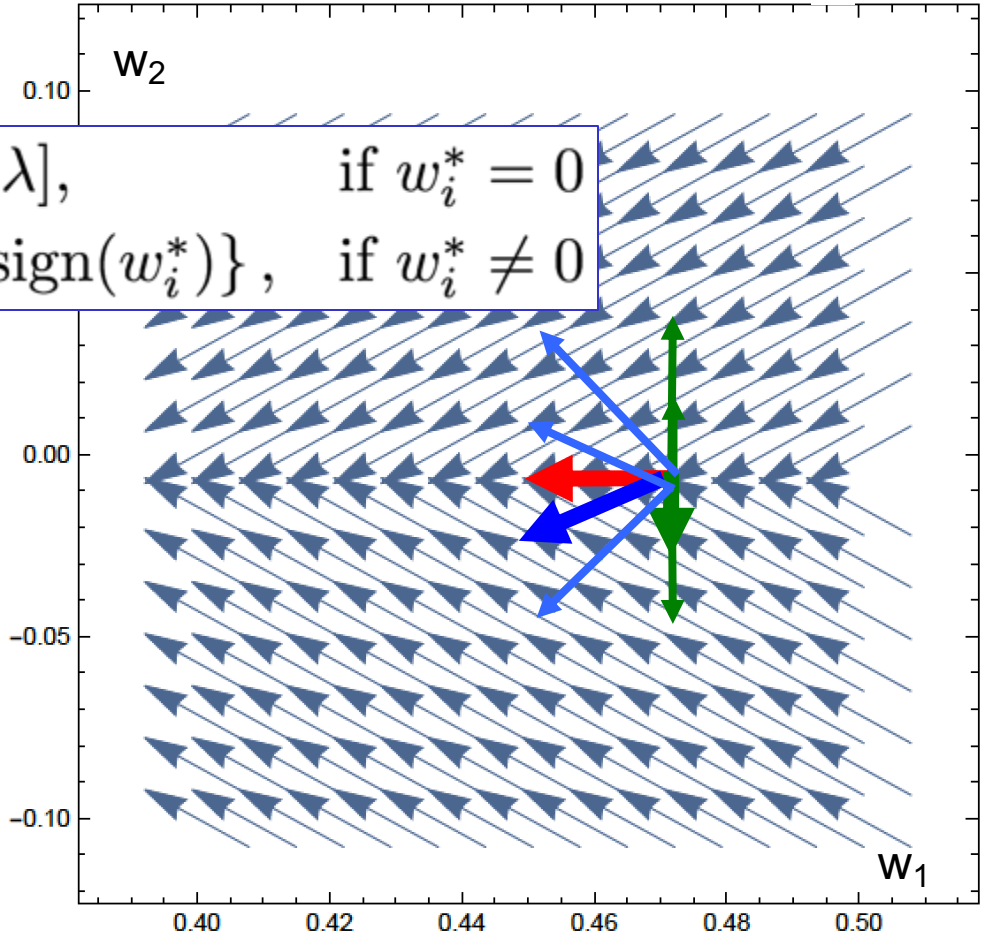
Minimum:
w_1=0.465, w_2=0

Empirical risk at w_1=0.46, w_2=0:
Negated gradient (one)
Negated partial derivative for w_1
Negated partial derivative for w_2

L_1 penalty at w_1=0.46, w_2=0:
Subgradients (more then one)
Negated w_1 coordinate of subgradients = -λ
Negated w_2 coordinate of subgradients ∈ [-λ, λ]

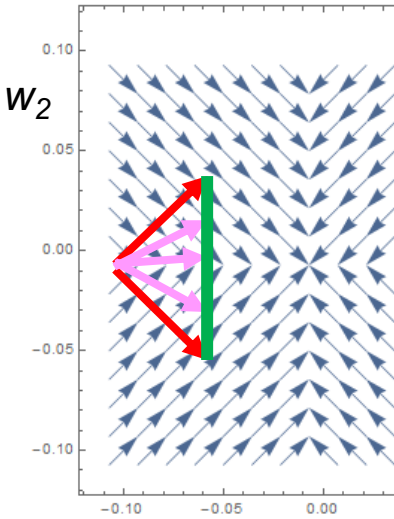
$$\frac{\partial \hat{R}_{S_m}(w^*)}{\partial w_i} \in \begin{cases} [-\lambda, \lambda], & \text{if } w_i^* = 0 \\ \{-\lambda \text{sign}(w_i^*)\}, & \text{if } w_i^* \neq 0 \end{cases}$$





Clarke subdifferential

- Subgradient / subdifferential above was defined for convex functions
- What about non-convex functions?
  - E.g. f(x)=-|x|
    - It seems it's not really that different from f(x)=|x|
- Clarke's generalized gradient



Definition 1 (Clarke, 1975 [8]). *The* generalized gradient $\partial f(x)$ *of a locally Lipschitz function* f *at* x *is defined as*

$$\partial f(x) = \text{conv}\left\{\lim_{x_k \to x} \nabla f(x_k)\right\},$$

where the limit is over all convergent sequences of x_k *where gradient exists, and* conv *denotes convex hull.*