DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 5 | Part 1

Introduction

Empirical Risk Minimization (ERM)

- Step 1: choose a **hypothesis class**
  - We've chosen linear predictors.
- Step 2: choose a **loss function**
- Step 3: find H minimizing **empirical risk**
  - In case of linear predictors, equivalent to finding $\vec{w}$.

Minimizing Empirical Risk

- We want to minimize the **empirical risk**:

$$R(\vec{w}) = \frac{1}{n}\sum_{i=1}^{n} \ell(H(\vec{x}^{(i)}; \vec{w}), y_i)$$

$$= \frac{1}{n}\sum_{i=1}^{n} \ell(\text{Aug}(\vec{x}^{(i)}) \cdot \vec{w}, y_i)$$

Minimizing Empirical Risk

- For some losses there's a formula for the best $\vec{w}$.
  - **Example:** square loss.
  - But it might be **too costly** to use!
- For others, there isn't.
  - **Example:** absolute loss, Huber loss.
- In either case, we might use **gradient descent**.

Two Issues with Gradient Descent

1. Can be **expensive** to compute the exact gradient.
   - Especially when we have a large data set.
   - **Solution: stochastic gradient descent**.

2. Doesn't work as-is if risk is **not differentiable**.
   - Such as with the absolute loss.
   - **Solution: subgradient descent**.

Today

- Answer two remaining questions:

1. How do we minimize the risk with respect to non-differentiable losses, like the **absolute loss**?

2. When is gradient descent **guaranteed** to work?

DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 5 | Part 2

Subgradient Descent

Gradient Descent?

- **Question**: can we use gradient descent if the risk is not differentiable?
- **Answer**: **yes**, with a slight modification.

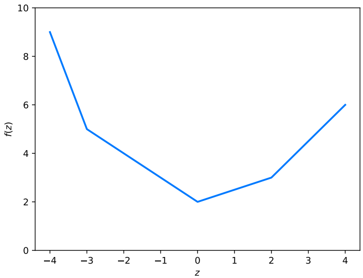
Differentiability

- A function $f(z)$ is **differentiable** if the derivative exists at every point.
- That is, it has a well-defined slope at every point.

Exercise

Where is the derivative **not** defined?

$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \leq z < 0 \\ 0.5z + 2 & \text{if } 0 \leq z < 2 \\ 3z/2 & \text{if } z \geq 2 \end{cases}$$

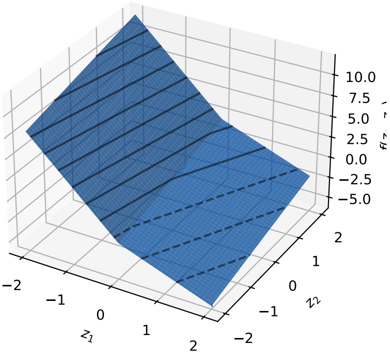


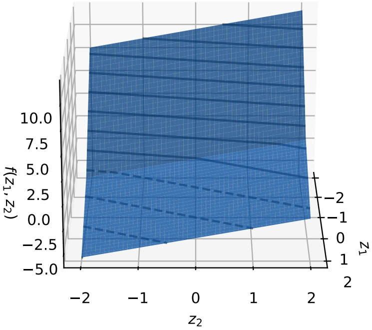
Differentiability

- A function $f(\vec{z})$ is **differentiable** if the **gradient** exists at every point.
- In other words, all of the slopes are well-defined:
  - $\partial f / \partial z_1, \partial f / \partial z_2, \ldots$

Example

- $f(z_1, z_2) = \begin{cases} -5z_1 + z_2 & \text{if } z_1 \leq 0 \\ -2z_1 + z_2 & \text{if } z_1 > 0 \end{cases}$

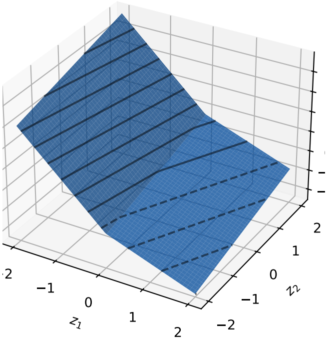




Exercise

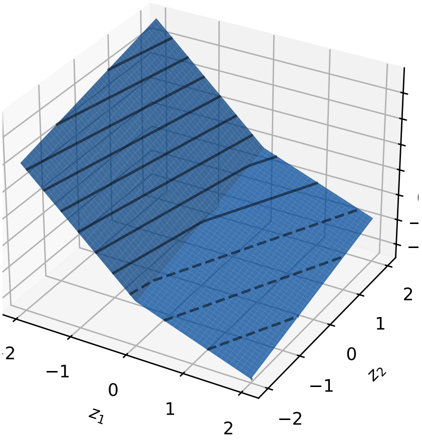
What is the gradient at (-1, -1)? (1, -1)? (0, 1)?

$$f(z_1, z_2) = \begin{cases} -5z_1 + z_2 & \text{if } z_1 \le 0 \\ -2z_1 + z_2 & \text{if } z_1 > 0 \end{cases}$$



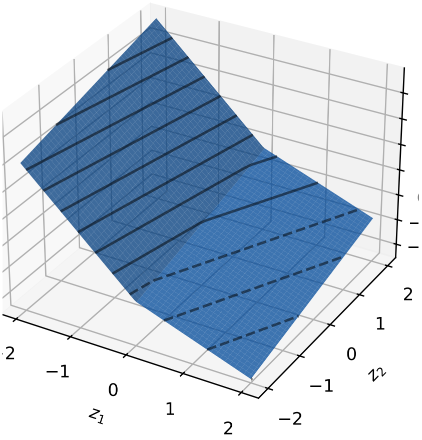
Answer

- $\vec{\nabla} f(\vec{z})$ is defined everywhere except along $z_1 = 0$.
- If $z_1 < 0$, $f(\vec{z}) = -5z_1 + z_2$.
  - gradient is $(-5, 1)^T$ here
- If $z_1 > 0$, $f(\vec{z}) = -2z_1 + z_2$.
  - gradient is $(-2, 1)^T$ here



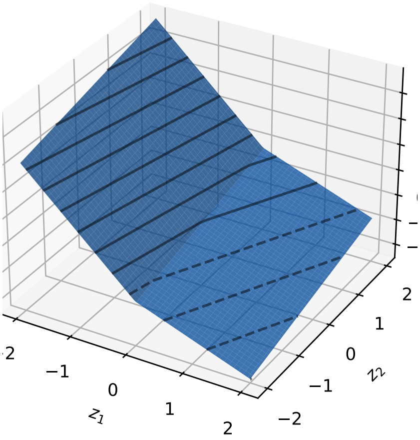
Answer

$$\frac{df}{d\vec{z}}(\vec{z}) = \begin{cases} (-5, 1)^T, & \text{if } z_1 < 0, \\ (-2, 1)^T, & \text{if } z_1 > 0, \\ \text{undefined}, & \text{if } z_1 = 0. \end{cases}$$



Problem

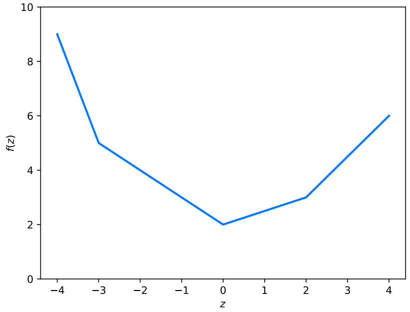
- We can try running gradient descent.
- But what do we do if we reach a point where the gradient is **not defined**?
- We need a **replacement** for the gradient that tells us where to go.



Idea

- Slope is undefined at $z_1 = -3$.
  - To the left, slope is -4
  - To the right, slope is -1

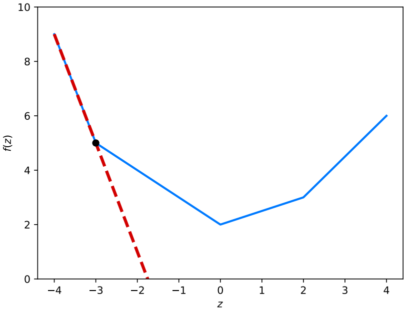
$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \leq z < 0 \\ 0.5z + 2 & \text{if } 0 \leq z < 2 \\ 3z/2 & \text{if } z \geq 2 \end{cases}$$



Idea

- Slope is undefined at $z_1 = -3$.
  - To the left, slope is -4
  - To the right, slope is -1

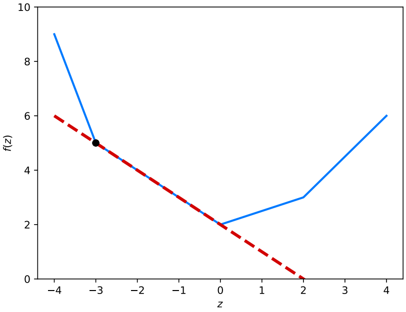
$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \le z < 0 \\ 0.5z + 2 & \text{if } 0 \le z < 2 \\ 3z/2 & \text{if } z \ge 2 \end{cases}$$



Idea

- Slope is undefined at $z_1 = -3$.
  - To the left, slope is -4
  - To the right, slope is -1

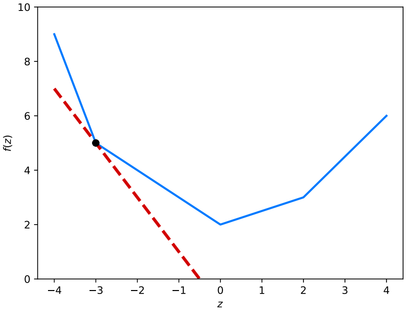
$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \leq z < 0 \\ 0.5z + 2 & \text{if } 0 \leq z < 2 \\ 3z/2 & \text{if } z \geq 2 \end{cases}$$



Idea

- Slope is undefined at $z_1 = -3$.
  - To the left, slope is -4
  - To the right, slope is -1

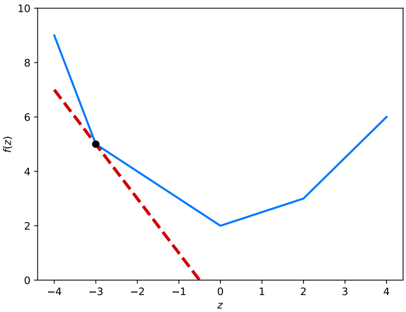
$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \le z < 0 \\ 0.5z + 2 & \text{if } 0 \le z < 2 \\ 3z/2 & \text{if } z \ge 2 \end{cases}$$



Idea

- Any number between -4 and -1 adequately describes the behavior of f at $z = -3$.

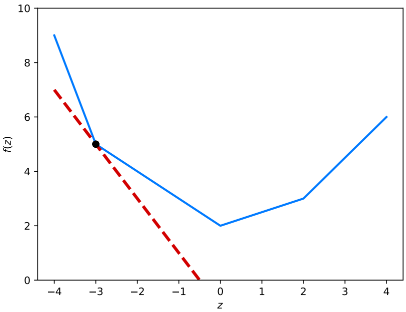
$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \le z < 0 \\ 0.5z + 2 & \text{if } 0 \le z < 2 \\ 3z/2 & \text{if } z \ge 2 \end{cases}$$



Idea

- Any number between -4 and -1 is a **subderivative** of f at $z = -3$.

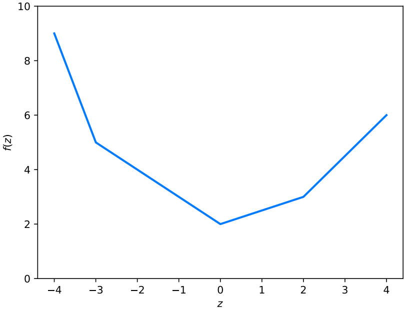
$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \le z < 0 \\ 0.5z + 2 & \text{if } 0 \le z < 2 \\ 3z/2 & \text{if } z \ge 2 \end{cases}$$



Exercise

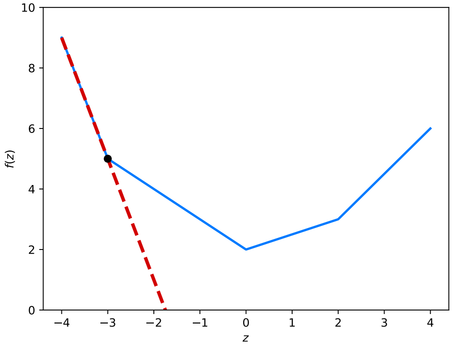
What are the valid subderivatives of f at $z = 2$?

$$f(z) = \begin{cases} -4z - 7 & \text{if } z < -3 \\ -z + 2 & \text{if } -3 \le z < 0 \\ 0.5z + 2 & \text{if } 0 \le z < 2 \\ 3z/2 & \text{if } z \ge 2 \end{cases}$$



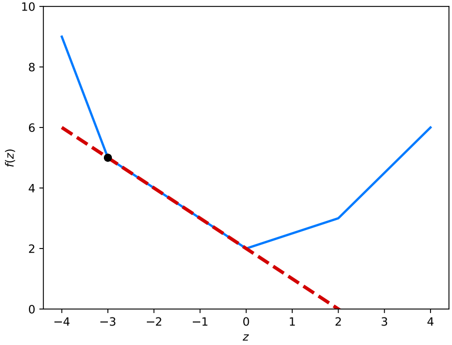
Subderivatives

- Any valid subderivative defines a line that lies below the function.



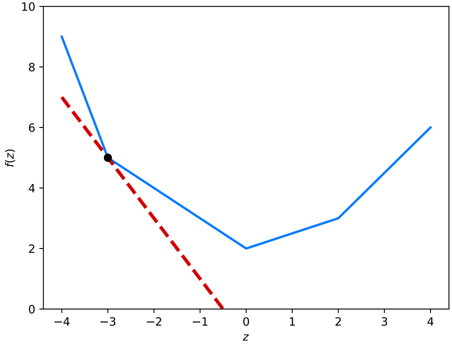
Subderivatives

- Any valid subderivative defines a line that lies below the function.



Subderivatives

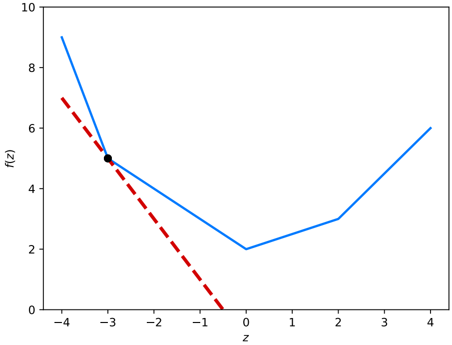
- Any valid subderivative defines a line that lies below the function.



Subderivatives

- The equation of this line is:

$$f_s(z) = f(z_0) + s(z - z_0)$$



Subderivatives

- A number s is a subderivative of f at z_0 if:

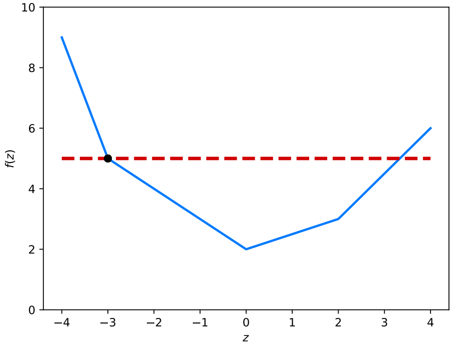
$$f(z) \geq f_s(z) \quad \text{for all } z$$

- That is, if:

$$f(z) \geq f(z_0) + s(z - z_0)$$

Exercise

Is 0 a valid subderivative of f at $z = 2$?



Intuition

- The **subderivative** tells us how the function changes when the slope doesn't exist.
- We can sometimes use it in place of a derivative.

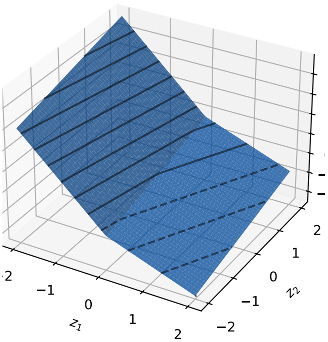
Subgradient

- In higher dimensions, we have multiple slopes to worry about.
- We can use a **subgradient** to generalize the concept of a subderivative.

Example

- There's no well-defined gradient at $z_1 = (0, 0)$.
  - The slope in the z_1 direction is undefined
  - Between -5 and -2?
  - The slope in the z_2 direction is 1

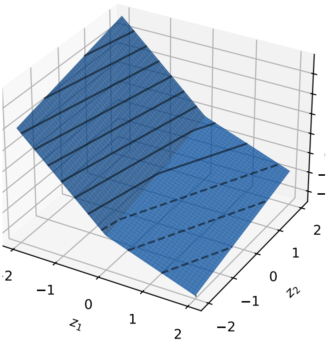
$$f(z_1, z_2) = \begin{cases} -5z_1 + z_2 & \text{if } z_1 \leq 0 \\ -2z_1 + z_2 & \text{if } z_1 > 0 \end{cases}$$



Example

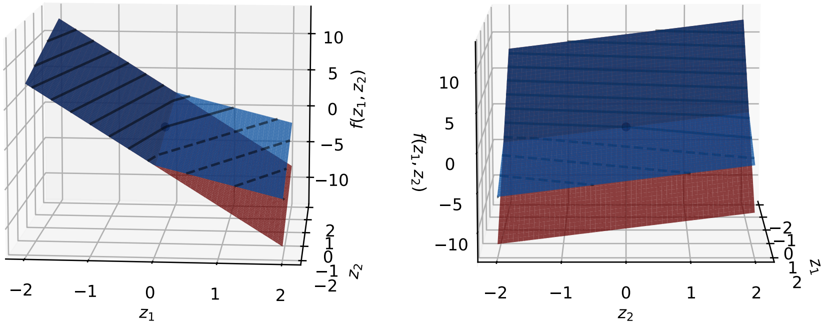
- We will call any vector $(s_1, 1)$ with $-5 \leq s_1 \leq -2$ a **subgradient** at $(0, 0)$.

$$f(z_1, z_2) = \begin{cases} -5z_1 + z_2 & \text{if } z_1 \leq 0 \\ -2z_1 + z_2 & \text{if } z_1 > 0 \end{cases}$$



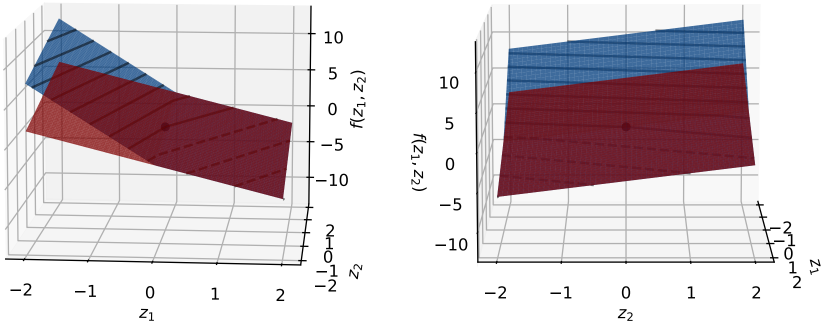
Subgradient

- A vector $\vec{s}$ defines a plane:
  - Example: $(-5, 1)^T$



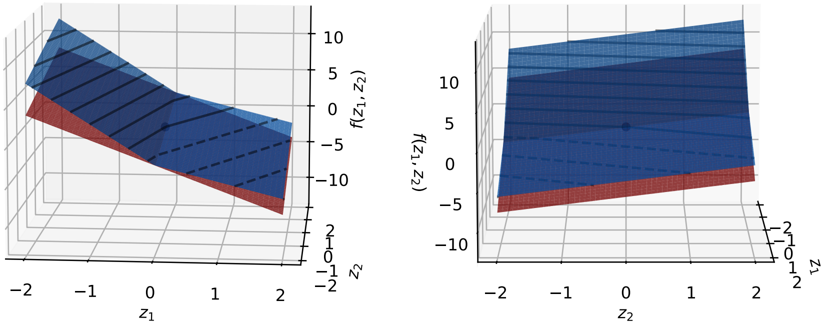
Subgradient

- A vector $\vec{s}$ defines a plane:
  - Example: $(-2, 1)^T$



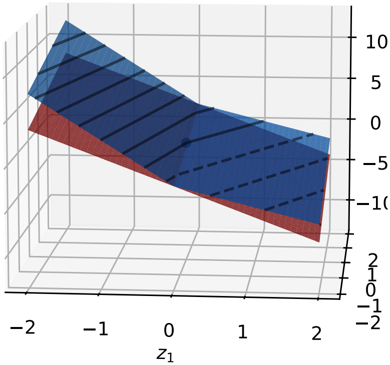
Subgradient

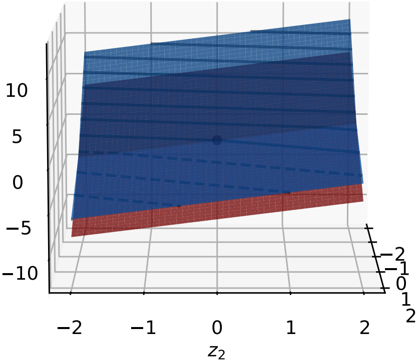
- A vector $\vec{s}$ defines a plane:
  - Example: $(-3, 1)^T$



Subgradient

- A vector $\vec{s}$ is a valid **subgradient** at $\vec{z}^{(0)}$ if the plane it defines lies at or below the function f.
  - Example: $(-3, 1)^T$

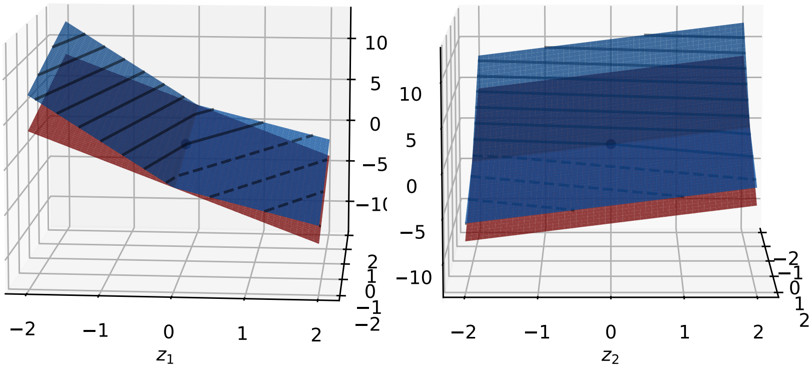




Subgradient

- The equation of the plane defined by $\vec{s}$ at $\vec{z}^{(0)}$ is:

$$f_s(\vec{z}) = f(\vec{z}^{(0)}) + \vec{s} \cdot (\vec{z} - \vec{z}^{(0)})$$



Subgradients

- $\vec{s}$ is a **subgradient** of $f(\vec{z})$ at $\vec{z}^{(0)}$ if:

$$f(\vec{z}) \geq f_s(\vec{z}) \quad \text{for all } \vec{z}$$

- That is, if:

$$f(\vec{z}) \geq f(\vec{z}^{(0)}) + \vec{s} \cdot (\vec{z} - \vec{z}^{(0)})$$

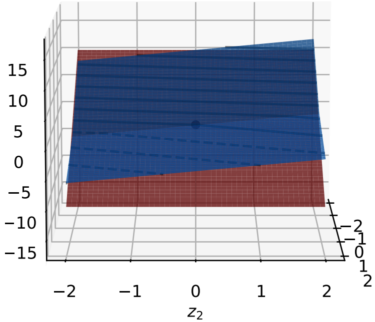
Finding Subgradients

- Here are two suggested ways to check that $\vec{s}$ is a valid subgradient.
- 1) Visualize it.
- 2) Check if the inequality holds.

Example

$$f(z_1, z_2) = \begin{cases} -5z_1 + z_2 & \text{if } z_1 \leq 0 \\ -2z_1 + z_2 & \text{if } z_1 > 0 \end{cases}$$

- Is $(-5, 0)^T$ a valid subgradient?



Example

$$f(z_1, z_2) = \begin{cases} -5z_1 + z_2 & \text{if } z_1 \leq 0 \\ -2z_1 + z_2 & \text{if } z_1 > 0 \end{cases}$$

- Is $(-5, 0)^T$ a valid subgradient at the point (0,0)?
- Is $f(0,0) + (-5,0)^T \cdot ((z_1, z_2) - (0,0)^T) \leq f(z_1, z_2) \quad$ for all z_1, z_2?

Tip

- If the slope is defined in a direction, the corresponding entry of the subgradient must be that slope.

Intuition

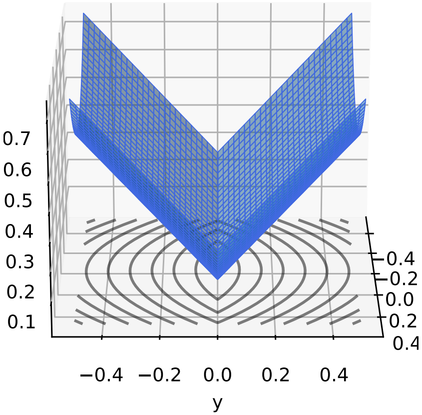
- A **subgradient** tells us where to go when the gradient is undefined.
- We can use it instead of the gradient in gradient descent.

Example

- $f(z_1, z_2) = z_1^2 + |z_2|$

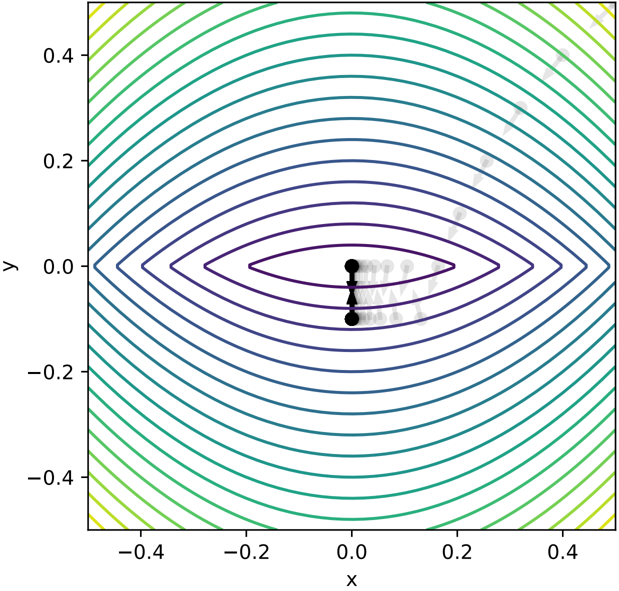
- A subgradient:

$$\vec{s}(z_1, z_2) = \begin{cases} (2z_1, 1)^T & \text{, if } z_2 > 0, \\ (2z_1, -1)^T & \text{, if } z_2 < 0, \\ (2z_1, 0)^T & \text{, if } z_2 = 0. \end{cases}$$



Example

- Subgradient descent on $f(z_1, z_2) = z_1^2 + |z_2|$
- Starting point: $(1/2, 1/2)^T$
- Learning rate: $\eta = 0.1$.



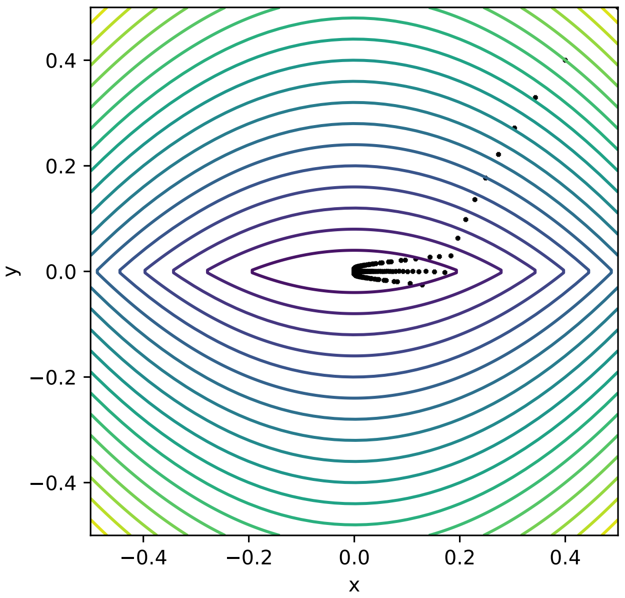
0.4
0.2
0.0
−0.2
−0.4
y
−0.4
−0.2
0.0
0.2
0.4
x

Problem

- Does not converge! Why?
- If f is differentiable, gradient gets smaller as we approach the minimum.
  - Naturally take smaller steps.
- Not true if the function is not differentiable!
  - Steps may stay the same size (too large).

Fix

- Decrease learning rate with each iteration.
- That is, choose a decreasing **learning rate schedule** $\eta(t) > 0$.
- **Theory:** choose $\eta(t) = c/\sqrt{t}$, where t is iteration #, c is a positive constant.


0.4
0.2
0.0
−0.2
−0.4
y
−0.4
−0.2
0.0
0.2
0.4
x

Subgradient Descent

To minimize $f(\vec{z})$:

- Pick arbitrary starting point $\vec{z}^{(0)}$, a decreasing **learning rate schedule** $\eta(t) > 0$.
- Until convergence, repeat:
  - **Compute a subgradient** $\vec{s}$ of f at $\vec{z}^{(i)}$.
  - Update $\vec{z}^{(t+1)} = \vec{z}^{(t)} - \eta(t)\vec{s}$
- When converged, return $\vec{z}^{(t)}$.

DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 5 | Part 3

Minimizing Risk w.r.t. Absolute Loss

Absolute Loss

- The **absolute loss** is a natural first choice for regression.

- The empirical risk becomes:

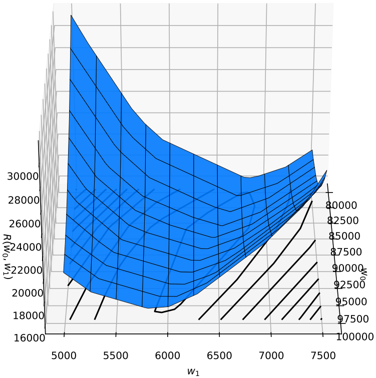
$$R_{\text{abs}}(\vec{w}) = \frac{1}{n}\sum_{i=1}^{n} |H(\vec{x}^{(i)}) - y_i|$$
$$= \frac{1}{n}\sum_{i=1}^{n} |\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) - y_i|$$

Minimizing the Risk

$$R(\vec{w}) = \frac{1}{n} \sum_{i=1}^{n} |\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) - y_i|$$

- We might try computing the gradient, setting to zero, and solving.
- But the risk is **not differentiable**.

Risk for the Absolute Loss



Regression with Absolute Loss

- We were stuck before.
    - This risk is **not differentiable**.

- **Now:** we can minimize the risk with respect to the absolute loss using **subgradient descent**.

Subgradient of Empirical Risk

- We need a **subgradient** of the empirical risk with respect to the absolute loss.

- **Useful fact**: the subgradient of a sum is the sum of the subgradients.[1]

- So it suffices to find a subgradient of the loss function:

$$\text{subgrad}\, R(\vec{w}) = \frac{1}{n} \sum_{i=1}^{n} \text{subgrad}\, \ell(\vec{w}; \vec{x}^{(i)}, y_i)$$

---

[1] At least, for convex functions.

Subgradient of the Absolute Loss

- We need a subgradient of the absolute loss.

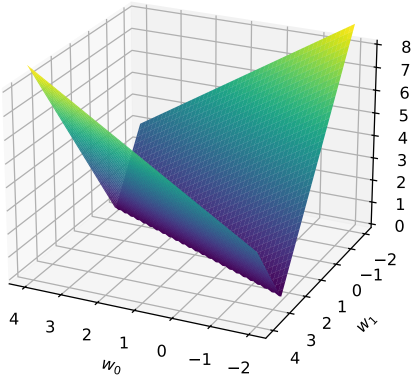
$$\ell_{\text{abs}}(\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}), y_i) = |\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) - y_i|$$

- An equivalent piecewise definition:

$$\ell_{\text{abs}}(\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}), y_i) = \begin{cases} \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) - y_i, & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) > y_i, \\ y_i - \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) < y_i, \\ 0, & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) = y_i. \end{cases}$$

The Absolute Loss

- Gradient exists except at $\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) = y_i$.
  - Here, we need a subgradient.



Exercise

What is the gradient when $\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) > y_i$? What about when $\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) < y_i$?

$$\ell_{\text{abs}}(\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}), y_i) = \begin{cases} \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) - y_i, & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) > y_i, \\ y_i - \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) < y_i, \\ 0, & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) = y_i. \end{cases}$$

Subgradient of the Absolute Loss

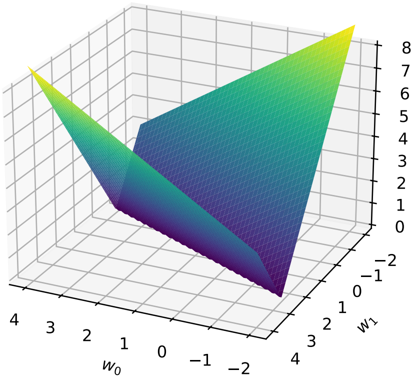
$$\ell_{\text{abs}}(\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}), y_i) = |\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) - y_i|$$

If $\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) > y_i$:
- Loss is $\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) - y_i$.
- Gradient is $\text{Aug}(\vec{x}^{(i)})$.

If $\vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) < y_i$:
- Loss is $y_i - \vec{w} \cdot \text{Aug}(\vec{x}^{(i)})$.
- Gradient is $-\text{Aug}(\vec{x}^{(i)})$.

Subgradient of the Absolute Loss

- The zero vector works as a subgradient.



Subgradient of the Absolute Loss

- Our subgradient of the absolute loss:

$$s(\vec{w}; \vec{x}^{(i)}, y_i) = \begin{cases} \text{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) > y_i, \\ -\text{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) < y_i, \\ \vec{0}, & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) = y_i. \end{cases}$$

Minimizing the Absolute Loss

- The subgradient of the empirical risk is the average of the subgradients of the loss:

$$
\begin{aligned}
&\text{subgrad. of } R(\vec{w}) \\
&\qquad = \frac{1}{n} \sum_{i=1}^{n} s(\vec{w}, \vec{x}^{(i)}, y_i) \\
&\qquad = \frac{1}{n} \sum_{i=1}^{n} \begin{cases} \operatorname{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \operatorname{Aug}(\vec{x}^{(i)}) > y_i, \\ -\operatorname{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \operatorname{Aug}(\vec{x}^{(i)}) < y_i, \\ \vec{0}, & \text{if } \vec{w} \cdot \operatorname{Aug}(\vec{x}^{(i)}) = y_i. \end{cases}
\end{aligned}
$$

Subgradient Descent

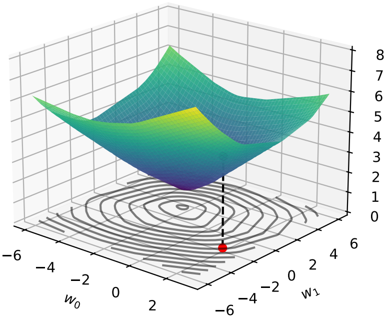
- We minimize the empirical risk with respect to the absolute loss using subgradient descent.

- Pick an initial $\vec{w}^{(0)}$, a decreasing learning rate schedule $\eta(t) > 0$.

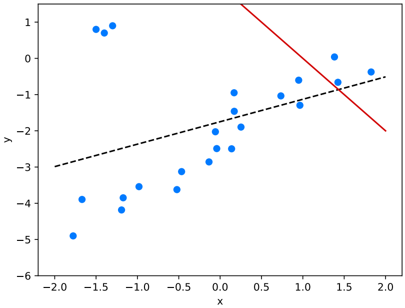
- Until convergence, repeat:
  - Update

$$\vec{w}^{(t+1)} = \vec{w}^{(t)} - \eta(t) \times \frac{1}{n} \sum_{i=1}^{n} \begin{cases} \text{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) > y_i, \\ -\text{Aug}(\vec{x}^{(i)}), & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) < y_i, \\ \vec{0}, & \text{if } \vec{w} \cdot \text{Aug}(\vec{x}^{(i)}) = y_i. \end{cases}$$

Example

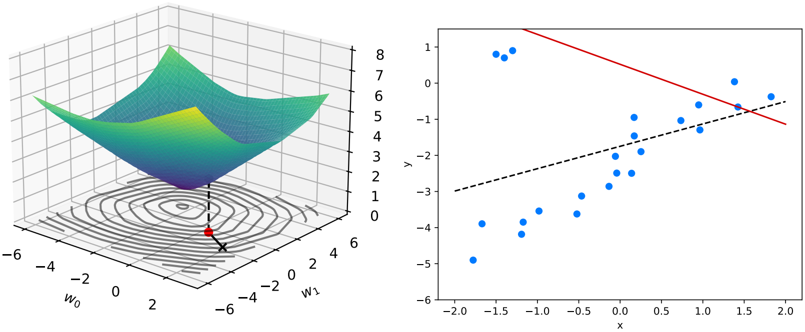


8
7
6
5
4
3
2
1
0
−6
−4
−2
0
2
w_0
6
4
2
0
−2
−4
−6
w_1



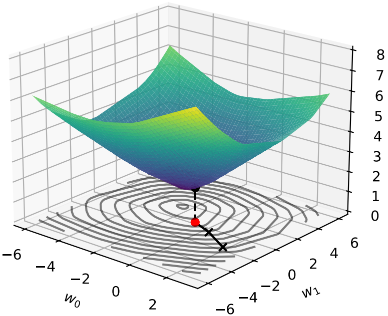
1
0
−1
−2
−3
−4
−5
−6
y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x

Example

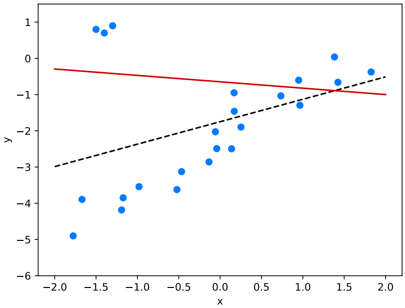


8
7
6
5
4
3
2
1
0
−6
−4
−2
0
2
w_0
−6
−4
−2
0
2
4
6
w_1
1
0
−1
−2
−3
−4
−5
−6
y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x

Example

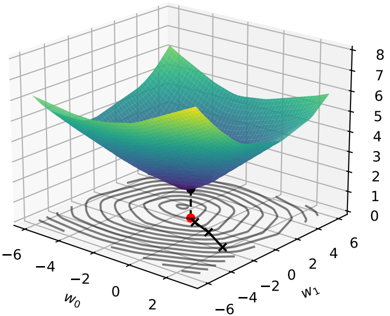


8
7
6
5
4
3
2
1
0
−6
−4
−2
0
2
w_0
−6
−4
−2
0
2
4
6
w_1

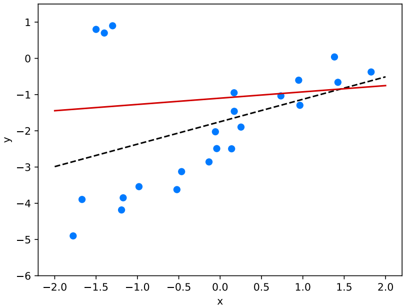


1
0
−1
−2
−3
−4
−5
−6
y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x

Example

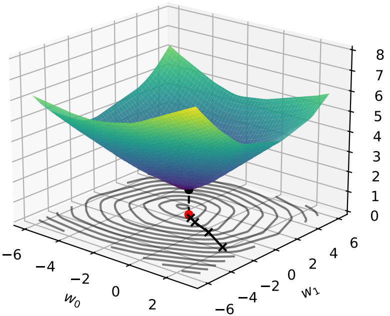


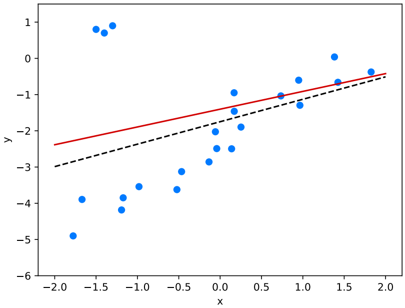
8
7
6
5
4
3
2
1
0
−6
−4
−2
0
2
w0
−6
−4
−2
0
2
4
6
w1



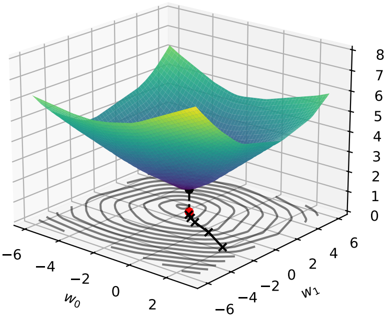
1
0
−1
−2
−3
−4
−5
−6
y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x

Example

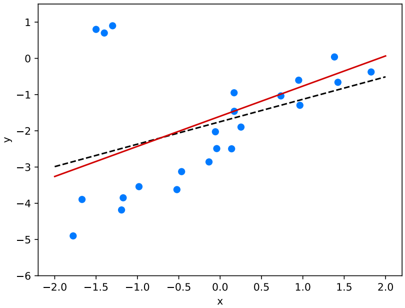




Example

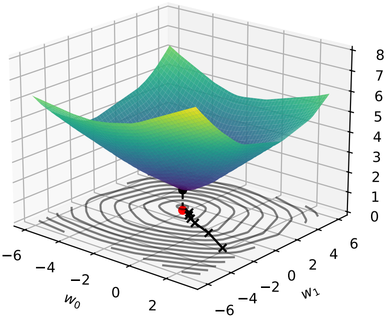


8
7
6
5
4
3
2
1
0
−6
−4
−2
0
2
w_0
−6
−4
−2
0
2
4
6
w_1

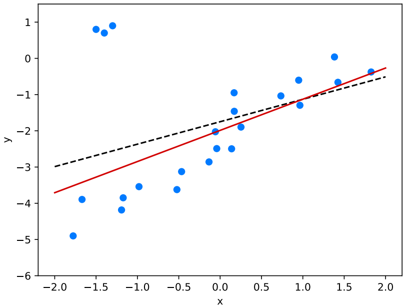


1
0
−1
−2
−3
−4
−5
−6
y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x

Example

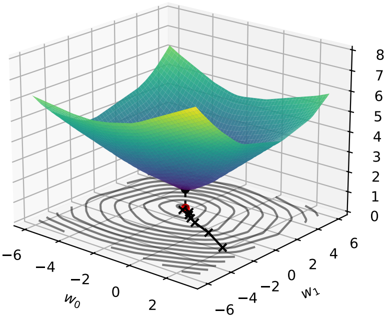


8
7
6
5
4
3
2
1
0
−6
−4
−2
0
2
w_0
6
4
2
0
−2
−4
−6
w_1

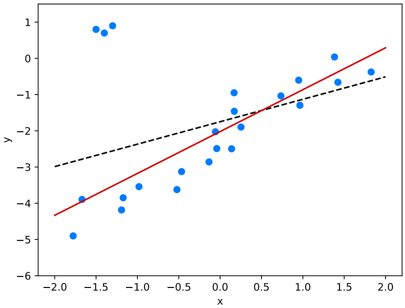


1
0
−1
−2
−3
−4
−5
−6
y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x

Example



8
7
6
5
4
3
2
1
0
−6
−4
−2
0
2
w_0
6
4
2
0
−2
−4
−6
w_1



1
0
−1
−2
−3
−4
−5
−6
y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x

In Practice

- We've minimized the risk with respect to the absolute loss.
- This approach has different names:
  - Quantile regression, median regression
  - Minimum Absolute Deviations (MAD)
- Solvable by (S)GD, or as a **linear program**.

DSC 140A

Probabilistic Modeling & Machine Learning

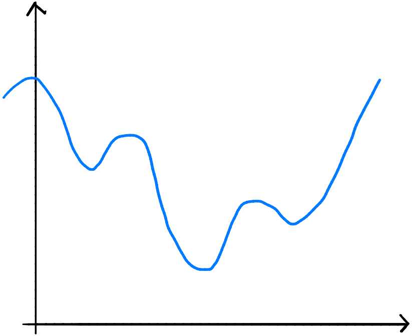
Lecture 5 | Part 4

Convexity

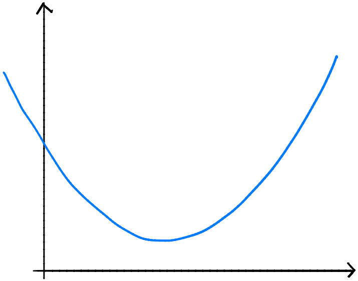
Question

- When is gradient descent guaranteed to work?

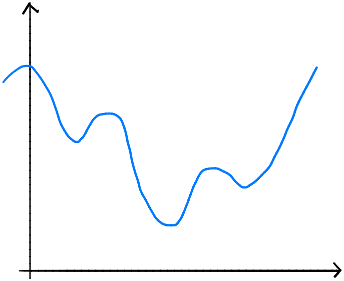
Not here...



Convex Functions



Convex



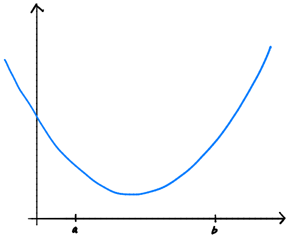
Non-convex

Convexity: Definition

- f is **convex** if for **every** a, b the line segment between

$$(a, f(a)) \qquad \text{and} \qquad (b, f(b))$$

does not go below the plot of f.

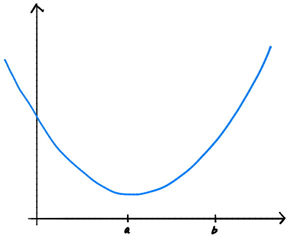


Convexity: Definition

- f is **convex** if for **every** a, b the line segment between

$$(a, f(a)) \qquad \text{and} \qquad (b, f(b))$$

does not go below the plot of f.

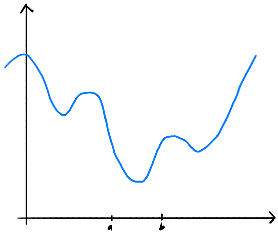


Convexity: Definition

- f is **convex** if for **every** a, b the line segment between

$$(a, f(a)) \qquad \text{and} \qquad (b, f(b))$$

does not go below the plot of f.

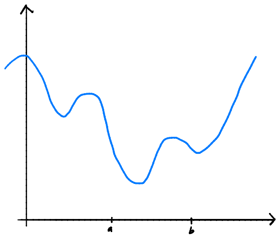


Convexity: Definition

- f is **convex** if for **every** a, b the line segment between

$$(a, f(a)) \qquad \text{and} \qquad (b, f(b))$$

does not go below the plot of f.



Other Terms

- If a function is not convex, it is **non-convex**.
- **Strictly convex**: the line lies strictly above curve.
- **Concave**: the line lies on or below curve.

Exercise

True or **False**: a convex function must have a unique global minimum.

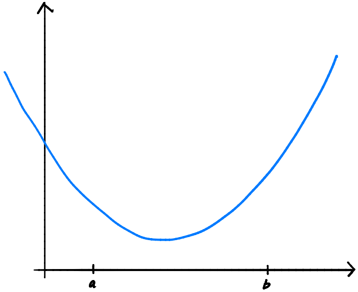
True or **False**: a local minimum of a convex function is always a global minimum.

True or **False**: a *strictly* convex function must have a unique global minimum.

Convexity: Formal Definition

- A function $f : \mathbb{R} \to \mathbb{R}$ is **convex** if for every choice of $a, b \in \mathbb{R}$ and $t \in [0, 1]$:

$$(1 - t)f(a) + tf(b) \geq f((1 - t)a + tb).$$

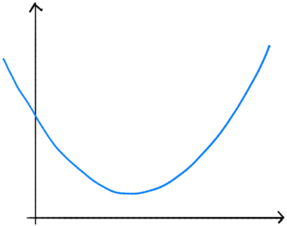


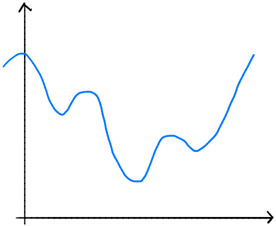
Exercise

Using the definition, is $f(x) = |x|$ convex?

Another View: Second Derivatives

- If $\frac{d^2 f}{dx^2}(x) \geq 0$ for all x, then f is convex.
- Example: $f(x) = x^4$ is convex.
- **Warning!** Only works if f is twice differentiable!





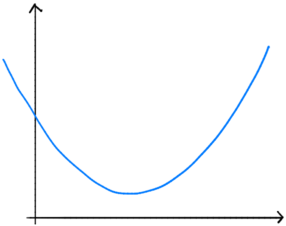
Another View: Second Derivatives

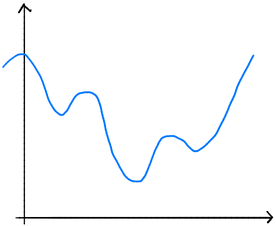
- "Best" straight line at x_0:
  - $f_1(x) = f(x_0) + f'(x_0) \cdot (x - x_0)$

- "Best" parabola at x_0:
  - $f_2(x) = f(x_0) + f'(x_0) \cdot (x - x_0) + \frac{1}{2} f''(x_0) \cdot (x - x_0)^2$
  - Possibilities: upward-facing, downward-facing, flat.

Convexity and Parabolas

- Convex if for **every** x_0, parabola is upward-facing (or flat).
  - That is, $f''(x_0) \geq 0$.





Proving Convexity Using Properties

Suppose that $f(x)$ and $g(x)$ are convex. Then:

- $w_1 f(x) + w_2 g(x)$ is convex, provided $w_1, w_2 \geq 0$
  - Example: $3x^2 + |x|$ is convex
- $g(f(x))$ is convex, provided g is non-decreasing.
  - Example: e^{x^2} is convex
- $\max\{f(x), g(x)\}$ is convex
  - Example: $\begin{cases} 0, & \text{x} < 0 \\ x, & \text{x} \geq 0 \end{cases}$ is convex

Note!

- These properties are useful for proving convexity for functions of **one variable**.
- Some of them will not generalize to higher dimensions.

Convexity and Gradient Descent

- Convex functions are (relatively) easy to optimize.
- **Theorem**: if $f(x)$ is convex and "not too steep"[2] then (stochastic) (sub)gradient descent converges to a **global optimum** of f *provided* that the step size is small enough[3]

[2]Technically, c-Lipschitz

[3]step size related to steepness, should decrease like $1/\sqrt{\text{step } \#}$.

DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 5 | Part 5

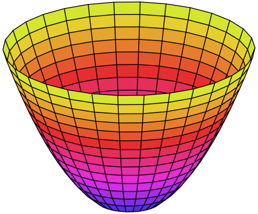
Convexity in Many Dimensions

Convexity: Definition

- $f(\vec{x})$ is **convex** if for **every** $\vec{a}, \vec{b}$ the line segment between

$$(\vec{a}, f(\vec{a})) \qquad \text{and} \qquad (\vec{b}, f(\vec{b}))$$

does not go below the plot of f.



Convexity: Formal Definition

- A function $f : \mathbb{R}^d \to \mathbb{R}$ is **convex** if for every choice of $\vec{a}, \vec{b} \in \mathbb{R}^d$ and $t \in [0, 1]$:

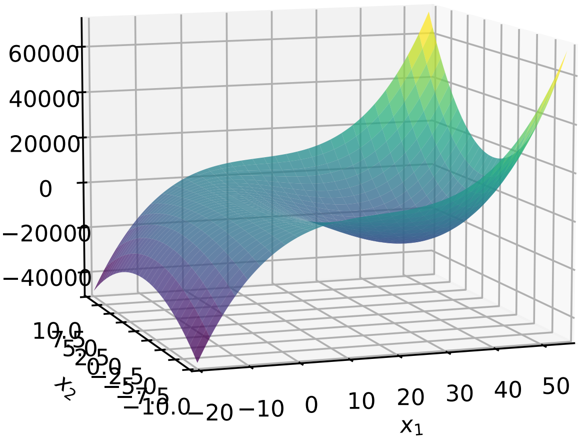
$$(1 - t)f(\vec{a}) + tf(\vec{b}) \geq f((1 - t)\vec{a} + t\vec{b}).$$

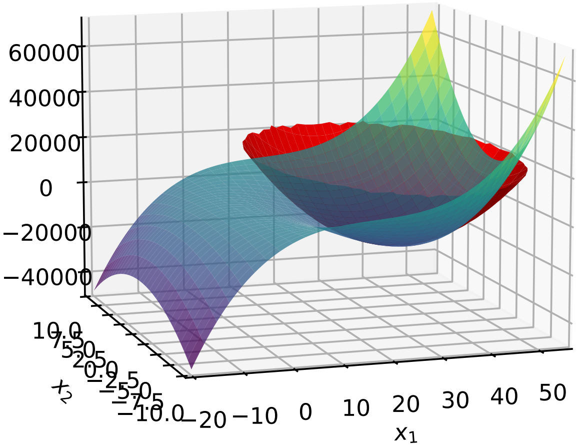
The Second Derivative Test

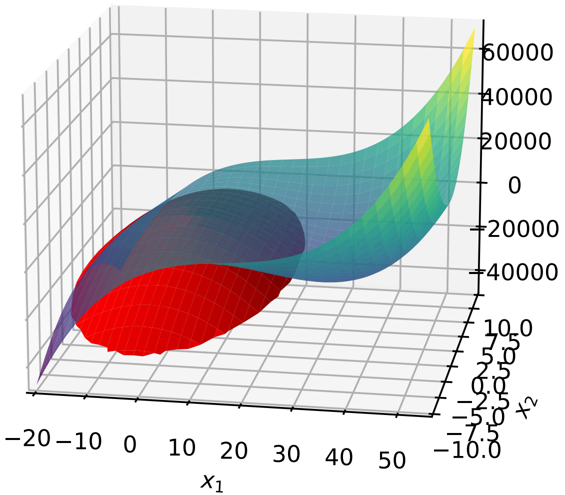
- For 1-dimensions functions:
  - convex if second derivative ≥ 0.
- For d-dimensional functions:
  - convex if ???

Second Derivatives in d-Dimensions

- In 2-dimensions, there are 4 second derivatives:
  - $\frac{\partial f^2}{\partial x_1^2}, \frac{\partial f^2}{\partial x_2^2}, \frac{\partial f^2}{\partial x_1 x_2}, \frac{\partial f^2}{\partial x_2 x_1}$
- In d-dimensions, there are d^2:
  - $\frac{\partial f^2}{\partial x_i \partial x_j}$ for all i, j.
- The second derivatives describe the curvature of a paraboloid approximating f.


60000
40000
20000
0
−20000
−40000
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
x_2
−20
−10
0
10
20
30
40
50
x_1


60000
40000
20000
0
−20000
−40000
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
x_2
−20
−10
0
10
20
30
40
50
x_1



60000
40000
20000
0
−20000
−40000
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
x_2
−20
−10
0
10
20
30
40
50
x_1

The Hessian Matrix

- Create the **Hessian** matrix of second derivatives:
- For $f : \mathbb{R}^2 \rightarrow \mathbb{R}$:

$$H(\vec{x}) = \begin{pmatrix} \frac{\partial f^2}{\partial x_1^2}(\vec{x}) & \frac{\partial f^2}{\partial x_1 x_2}(\vec{x}) \\ \frac{\partial f^2}{\partial x_2 x_1}(\vec{x}) & \frac{\partial f^2}{\partial x_2^2}(\vec{x}) \end{pmatrix}$$

In General

- If $f : \mathbb{R}^d \to \mathbb{R}$, the **Hessian** at $\vec{x}$ is:

$$H(\vec{x}) = \begin{pmatrix} \frac{\partial f^2}{\partial x_1^2}(\vec{x}) & \frac{\partial f^2}{\partial x_1 x_2}(\vec{x}) & \cdots & \frac{\partial f^2}{\partial x_1 x_d}(\vec{x}) \\ \frac{\partial f^2}{\partial x_2 x_1}(\vec{x}) & \frac{\partial f^2}{\partial x_2^2}(\vec{x}) & \cdots & \frac{\partial f^2}{\partial x_2 x_d}(\vec{x}) \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f^2}{\partial x_d x_1}(\vec{x}) & \frac{\partial f^2}{\partial x_d^2}(\vec{x}) & \cdots & \frac{\partial f^2}{\partial x_d^2}(\vec{x}) \end{pmatrix}$$

Second Derivative Test

- A function $f : \mathbb{R}^d \to \mathbb{R}$ is **convex** if for any $\vec{x} \in \mathbb{R}^d$, all **eigenvalues** of the Hessian matrix $H(\vec{x})$ are ≥ 0.

For This Class...

- You will not need to compute eigenvalues “by hand”...
- Unless the matrix is diagonal.
  - In which case, the eigenvalues are the diagonal entries.

Example

- The eigenvalues of this matrix are 5, 2, and 1.

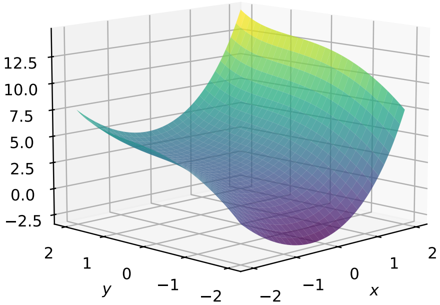
$$\begin{pmatrix} 5 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Exercise

Is $f(x, y) = e^x + e^y + x^2 - y^2$ convex?

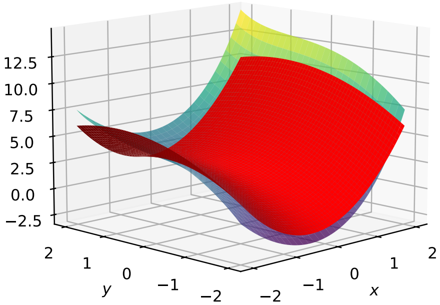
No

- The Hessian at (0,0) has a negative eigenvalue.



No

- The Hessian at (0,0) has a negative eigenvalue.



Exercise

Is $f(\vec{w}) = \|\vec{w}\|^2$ convex?

Note

- The second derivative test only works if f is twice differentiable.

- A function can be convex without having a second derivative.

Properties

- We can often prove convexity using properties.
- Two useful properties:
  - Sums of convex functions are convex.
  - Affine compositions of convex functions are convex.

Sums of Convex Functions

- Suppose that $f(\vec{x})$ and $g(\vec{x})$ are convex. Then $w_1 f(\vec{x}) + w_2 g(\vec{x})$ is convex, provided $w_1, w_2 \geq 0$.

Affine Composition

- Suppose that $f(x)$ is convex. Let A be a matrix, and $\vec{x}$ and $\vec{b}$ be vectors. Then

  $$g(\vec{x}) = f(A\vec{x} + \vec{b})$$

  is convex as a function of $\vec{x}$.

- **Remember:** a vector is a matrix with one column/row.

- Useful!

Exercise

Consider the function

$$f(\vec{w}) = (\vec{x} \cdot \vec{w} - y)^2$$

Is this function convex as a function of $\vec{w}$?

DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 5 | Part 6

Convex Loss Functions

Empirical Risk Minimization (ERM)

- Step 1: choose a **hypothesis class**
  - We've chosen linear predictors, $H(\vec{x}) = \text{Aug}(\vec{x}) \cdot \vec{w}$.
- Step 2: choose a **loss function**
- Step 3: find $\vec{w}$ minimizing **empirical risk**
  - Some choices of loss function make this **easier**.

Convexity and Gradient Descent

- Convex functions are (relatively) easy to optimize.
- **Theorem**: if $f(x)$ is convex and "not too steep"[4] then (stochastic) (sub)gradient descent converges to a **global optimum** of f *provided* that the step size is small enough[5].

---

[4] Technically, c-Lipschitz

[5] step size related to steepness, should decrease like $1/\sqrt{\text{step \#}}$

Convex Loss

- **Recall:** sums of convex functions are convex.
- **Implication:** if loss function is convex as a function of $\vec{w}$, so is the empirical risk, $R(\vec{w})$

$$R(\vec{w}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\operatorname{Aug}(\vec{x}^{(i)}) \cdot \vec{w}, y_i)$$

- **Takeaway:** Convex losses make ERM **easier**.

Example: Square Loss

- Recall the square loss for a linear predictor:

$$\ell_{\text{sq}}(\text{Aug}(\vec{x}) \cdot \vec{w}, y) = (\text{Aug}(\vec{x}) \cdot \vec{w} - y)^2$$

- This is **convex** as a function of $\vec{w}$.
- **Proof**: a few slides ago.

Example: Absolute Loss

- Recall the absolute loss for a linear predictor:

$$\ell_{\text{abs}}(\text{Aug}(\vec{x}) \cdot \vec{w}, y) = |\,\text{Aug}(\vec{x}) \cdot \vec{w} - y|$$

- This is **convex** as a function of $\vec{w}$.

Linear Predictors

- It's also important that we've chosen linear predictors.

- A loss that is **convex** in $\vec{w}$ for linear $H_1(x)$ may be **non-convex** for non-linear $H_2(x)$.

- Example: square loss.
  - If $H_1(x) = w_0 + w_1 x$, then $(w_0 + w_1 x - y)^2$ is **convex**.
  - If $H_2(x) = w_0 e^{w_1 x}$, then $(w_0 e^{w_1 x} - y)^2$ is **non-convex**.

Summary

- By combining 1) linear predictors and 2) a convex loss function, we make ERM **easier**.

- **Many** machine learning algorithms are linear predictors with convex loss functions.
  - As we'll see...

DSC 140A

Probabilistic Modeling & Machine Learning

Lecture 5 | Part 7

Appendix: From Theory to Practice

Gradient Descent

- We’ve spent three lectures on **gradient descent**.
- A powerful optimization algorithm.
- In practice, we use extensions of (stochastic) gradient descent.

Extensions of SGD

- Newton's method
  - Second order optimization, using the Hessian.
  - Can converge in fewer steps.
  - But the Hessian is **expensive** to compute.

- Adagrad, RMSprop, Adam
  - SGD with adaptive learning rates.
  - Used heavily in training of deep neural networks.

Non-Convex Optimization

- So far, we've only seen convex risks.
- But there's an important class of machine learning algorithms that have **non-convex** risks.
- **Namely:** deep neural networks.

Empirical Risk Minimization (ERM)

- Step 1: choose a **hypothesis class**
  - **Deep neural networks**.
- Step 2: choose a **loss function**
- Step 3: find $\vec{w}$ minimizing **empirical risk**

Deep Learning

- A **deep neural network** is a prediction function $H(\vec{x}; \vec{w})$ composed of many layers.
- Typically, H is not linear in $\vec{w}$.
- The risk becomes highly **non-convex**.
  - Even, for example, the square loss.
- How do we minimize the empirical risk?

Answer: SGD

- We use **stochastic gradient descent** (and extensions).
  - Even though the empirical risk is **non-convex**.
  - The optimization problem becomes much harder.

- SGD may not find a global minimum of the risk.

- But often finds a “**good enough**” local minimum.

Next Time

- Linear classification.