11.2 Convexity

Lecture based on "Dive into Deep Learning" **http://D2L.AI** (Zhang et al., 2020)

Prof. Dr. Christoph Lippert

Digital Health & Machine Learning

- A **set** X in a vector space is **convex** if for any $a, b \in X$ the line segment connecting a and b is also in X.
- For all $\lambda \in [0, 1]$ we have

$$\lambda \cdot a + (1 - \lambda) \cdot b \in X \text{ whenever } a, b \in X.$$

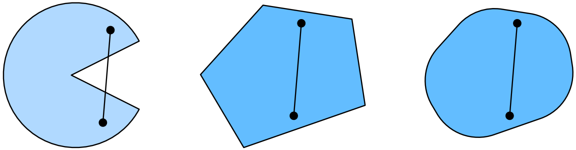


Figure: Three shapes, the left one is non-convex, the others are convex

Theorem

Assume that X and Y are convex sets.
Then $X \cap Y$ is also convex.

Proof.

Consider any $a, b \in X \cap Y$.
Since X and Y are convex, the line segments connecting a and b are contained in both X and Y.

Given that, they also need to be contained in $X \cap Y$. $\square$

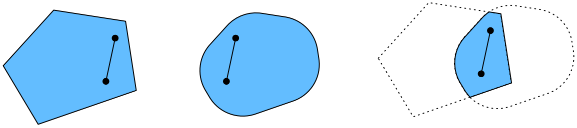
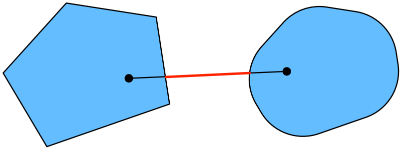


Figure: The intersection between two convex sets is convex

- Given convex sets X_i, their intersection $\cap_i X_i$ is convex.
- The converse is not true, consider two disjoint sets $X \cap Y = \emptyset$.
- Now pick $a \in X$ and $b \in Y$.
- The line segment connecting a and b needs to contain some part that is neither in X nor Y, since we assumed that $X \cap Y = \emptyset$.
- Hence the line segment isn't in $X \cup Y$ either, thus proving that in general unions of convex sets need not be convex.

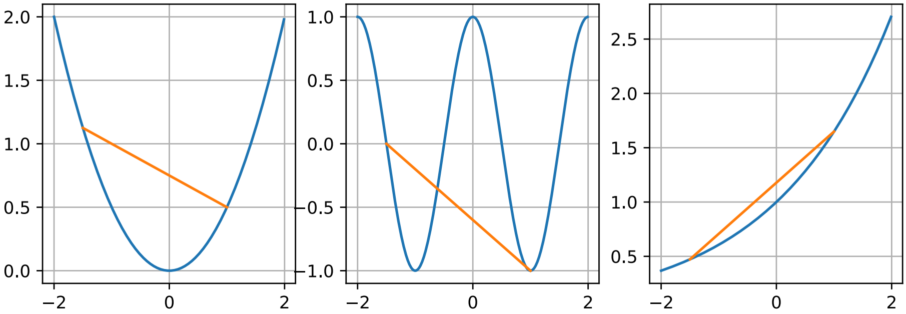


The union of two convex sets need not be convex.

Given a convex set X a **function** $f : X \to \mathbb{R}$ is **convex** if for all $x, x' \in X$ and for all $\lambda \in [0, 1]$ we have

$$\lambda f(x) + (1 - \lambda) f(x') \geq f(\lambda x + (1 - \lambda) x').$$

$\frac{1}{2}x^2$ $\qquad$ $\cos(\pi x)$ $\qquad$ $\exp\left(\frac{1}{2}x\right)$

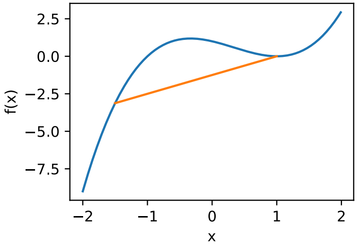


Convex functions have no local minima.

- if we minimize functions we cannot ‘get stuck’.
- There might be multiple global minima.

  For instance, $f(x) = \max(|x| - 1, 0)$ attains its minimum value over the interval $[-1, 1]$.
- No minimum may exist.

  $f(x) = \exp(x)$ does not attain a minimum value on $\mathbb{R}$. For $x \to -\infty$ it asymptotes to 0, however there is no x for which $f(x) = 0$.



$$f(x) = (x+1)(x-1)^2$$

- Local minimum for $x = 1$.
- No global minimum
- Not convex

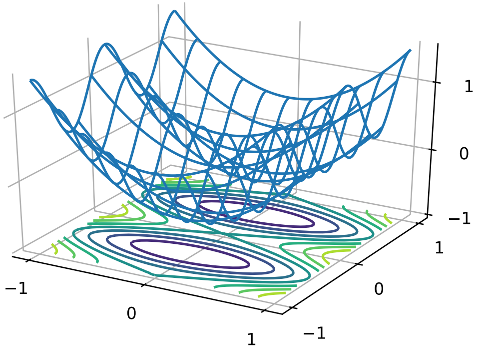
Convex functions f have convex **below-sets**.

$$S_b := \{x | x \in X \text{ and } f(x) \leq b\}.$$

- For any $x, x' \in S_b$ and $\lambda \in [0,1]$ we get due to convexity

$$f(\lambda x + (1-\lambda)x') \leq \lambda f(x) + (1-\lambda) f(x') \leq b$$

- Thus $\lambda x + (1-\lambda)x' \in S_b$.



$$g(x,y) = 0.5x^2 + \cos(2\pi y)$$

- g is non-convex.
- The **level sets** are non-convex.

Convex optimization allows us to handle **constraints** efficiently.

$$\underset{\mathbf{x}}{\text{minimize}}\ f(\mathbf{x})$$
$$\texttt{subject to}\ \ c_i(\mathbf{x}) \leq 0,\ \forall\, i \in \{1, \ldots, N\}$$

- f is the objective
- c_i are constraint functions

Examples:

- To constrain the parameters $\mathbf{x}$ to the unit ball:
$$c_1(\mathbf{x}) = \|\mathbf{x}\|_2 - 1$$
- To constrain all $\mathbf{x}$ to lie on a half-space:
$$c_2(\mathbf{x}) = \mathbf{v}^\top \mathbf{x} + b$$
- Satisfying both constraints simultaneously amounts to a slice of a ball as the constraint set.

Imagine a ball inside a box.

- The ball will roll to the place that is lowest and the forces of gravity will be balanced out with the forces that the sides of the box can impose on the ball.
- In short, the gradient of the objective function (i.e. gravity) will be offset by the gradient of the constraint function (need to remain inside the box by virtue of the walls ‘pushing back’).
- Note that any constraint that is not **active** (i.e. the ball doesn’t touch the wall) will not be able to exert any force on the ball.

The **Lagrange function** L implements this intuition:

$$L(\mathbf{x}, \alpha) = f(\mathbf{x}) + \sum_i \alpha_i c_i(\mathbf{x}) \text{ where } \alpha_i \geq 0$$

- The variables α_i are the **Lagrange Multipliers** ensuring that a constraint is properly enforced.
- They are chosen just large enough to ensure that $c_i(\mathbf{x}) \leq 0 \ \forall \ i$
- For any $\mathbf{x}$ for which $c_i(\mathbf{x}) < 0$ naturally, $\alpha_i = 0$

- Instead of solving the constrained problem exactly, in deep learning, we typically only aim for an approximate solution by adapting the Lagrange function L.
- Rather than satisfying $c_i(\mathbf{x}) \leq 0$ we simply add $\alpha_i c_i(\mathbf{x})$ to the objective function $f(x)$

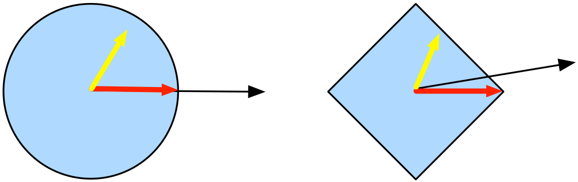
Example (weight decay)

- Add $\frac{\lambda}{2}\|\mathbf{w}\|^2$ to the objective function to ensure that $\mathbf{w}$ doesn't grow too large.
- This is equivalent to constrained optimization with $\|\mathbf{w}\|^2 - r^2 \leq 0$ for some radius r.
- Adjusting the value of λ allows us to vary the size of $\mathbf{w}$.

Projections on a (convex) set X are an alternative strategy for satisfying constraints.

$$\mathsf{Proj}_X(\mathbf{x}) = \operatorname*{arg\,min}_{\mathbf{x}' \in X} \|\mathbf{x} - \mathbf{x}'\|_2$$

The closest point in X to $\mathbf{x}$.



Example (Gradient Clipping)

- Ensure that a gradient has length bounded by c via

$$\mathbf{g} \leftarrow \mathbf{g} \cdot \min(1, \frac{c}{\|\mathbf{g}\|})$$

- This turns out to be a *projection* of $\mathbf{g}$ onto the ball of radius c.

- Points inside the set (yellow) remain unchanged.
- Points outside the set (black) are mapped to the closest point inside the set (red).
- For ℓ_2 balls the direction is unchanged.
- Projecting $\mathbf{w}$ onto an ℓ_1 ball produces sparse weight vectors.

Summary

In the context of deep learning the main purpose of convex functions is to motivate optimization algorithms and help us understand them in detail.

- Intersections of convex sets are convex.

  Unions are not.
- A twice-differentiable function is convex if and only if its second derivative has only non-negative eigenvalues throughout.
- Convex constraints can be added via the Lagrange function.

  In practice simply add them with a penalty to the objective function.
- Projections map to points in the (convex) set closest to the original point.