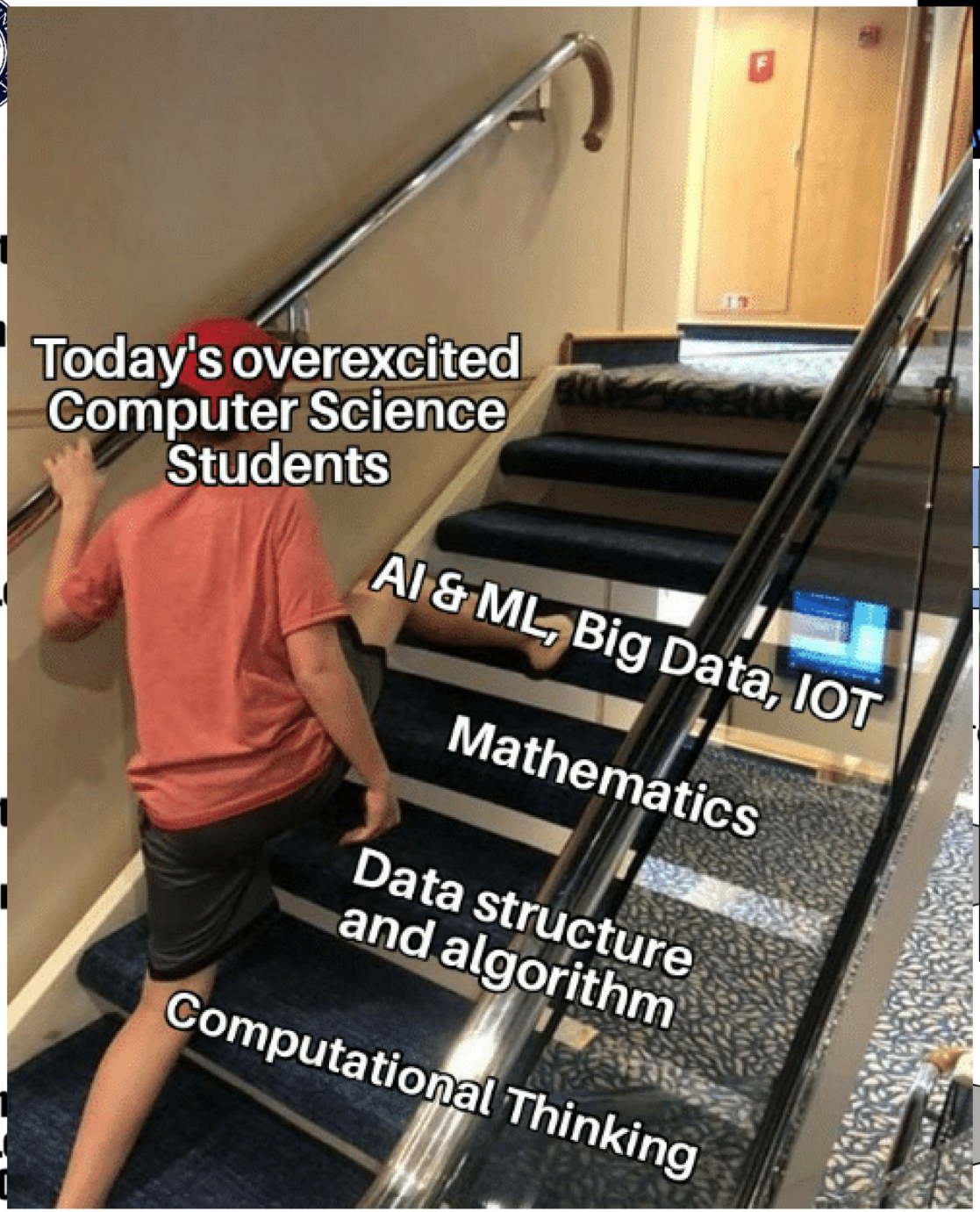
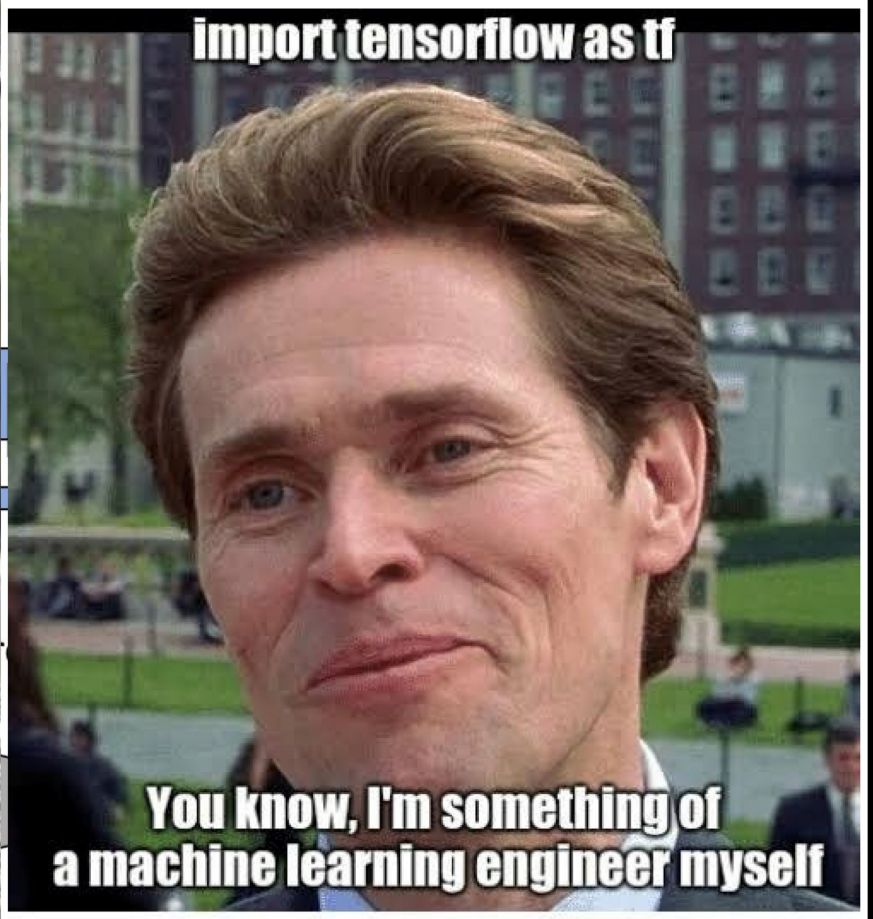
CSE 4820/5819 Introduction to Optimization

Suining He
Department of Computer Science and Engineering
University of Connecticut
suining.he@uconn.edu





You know...

$$\theta^* = \arg\min_{\theta}\{\sum_i \|w_i c_i\|^2\}$$

Theseus

Task

Loss

Reality is often disappointing

Optimization Problem: Definition

- Optimization Problem:
  - Determine value of optimization variable within feasible region/set to optimize optimization objective

$$\min_{x} f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

  - Optimization variable $x \in \mathbb{R}^n$
  - Feasible region/set $\mathcal{F} \subset \mathbb{R}^n$
  - Optimization objective $f: \mathcal{F} \to \mathbb{R}$

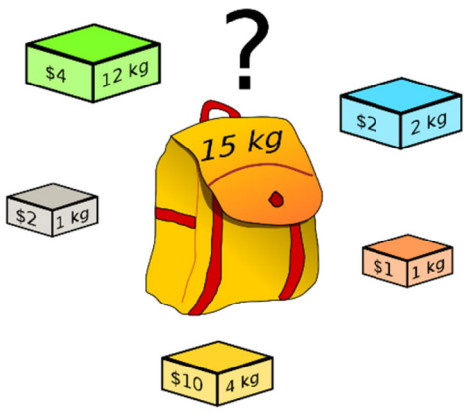
- Optimal solution: $x^* = \underset{x \in \mathcal{F}}{\operatorname{argmin}} f(x)$
- Optimal objective value $f^* = \min_{x \in \mathcal{F}} f(x) = f(x^*)$

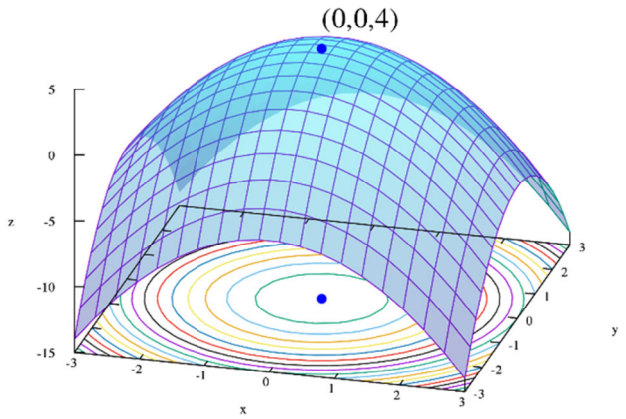
Optimization Problem: Definition

$$\min_{x} f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

- Optimization variable $x \in \mathbb{R}^n$
  - Discrete variables: Combinatorial optimization
  - Continuous variables: Continuous optimization
  - Mixed: Some variables are discrete, and some are continuous

Knapsack





Optimization Problem: Definition

$$\min_{x} f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

- Feasible region or set $\mathcal{F} \subset \mathbb{R}^n$
  - Unconstrained optimization: $\mathcal{F} = \mathbb{R}^n$
  - Constrained optimization: $\mathcal{F} \subsetneq \mathbb{R}^n$
    - Find a feasible point $x \in \mathcal{F}$ can already be difficult

Optimization Problem: Definition

$$\min_{x} f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

- Optimization objective $f: \mathcal{F} \to \mathbb{R}$
  - $f(x) = 1$: Feasibility problem
    - Simple functions
      - Linear function $f(x) = a^T x$
      - Convex function
    - Complicated functions
      - Even can be implicitly represented through an algorithm which takes $x \in \mathcal{F}$ as input, and outputs a value

Optimization Problem: Definition

$$\min_{x} f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

- Minimization can be converted to maximization (and vice versa)

$$\max_{x} g(x) = -f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

Same optimal solution

Optimal objective value $g^* = -f^*$

Why Do We Care?

- Optimization is at the heart of many (most practical?) machine learning algorithms
  - MLE $\underset{\theta}{\text{maximize}}\ \mathcal{L}(\theta|\mathbf{X}) = \log L(\theta|\mathbf{X}) = \sum_i \log p(x_i|\theta)$
  - Linear regression
  - Classification (logistic regression or SVM)
  - K-means clustering

$$\underset{\mu_1,\ldots,\mu_k}{\text{minimize}}\ J(\mu) = \sum_{j=1}^{k} \sum_{i \in C_j} \|x_i - \mu_j\|^2$$

Optimization Problem: Example

x_i	1.0	2.0	3.5
y_i	2.1	3.98	7.0

- Example: *Linear Regression*
  - *Problem*: Find w such that $y_i \approx wx_i$, $\forall i = 1, 2, 3$
    - Variables: w
    - Feasible region: $\mathbb{R}$
  - Objective function $f(w)$?

$$\min_{x} f(x) \quad \text{s.t. } x \in \mathcal{F}$$

$$\min_{w} \sum_{i=1}^{3} |y_i - wx_i| \quad \text{s.t. } w \in \mathbb{R}$$

$$\min_{w} \sum_{i=1}^{3} (y_i - wx_i)^2 \quad \text{s.t. } w \in \mathbb{R}$$

Optimization Problem

- Many algorithms developed for special classes of optimization problems (i.e., when $f(x)$ and $\mathcal{F}$ satisfy certain constraints
  - Convex optimization problem (CO)
  - Linear programming (LP)
  - Quadratic programming (QP)

Linear Programming & Quadratic Programming

- Formulation

$$\min_x A^T x$$
$$\text{s.t. } c_i^T x \leq b_i, i = 1, 2, \ldots, m$$

$c_1 \cdot x \leq b_1$

$=$ → # of constraints

- No exact formula for solution
  - Reliable and efficient algorithms and software
  - A mature technology

- Formulation

$$\min_x 1/2\, x^T P x + c^T x + d$$
$$\text{s.t. } g_i^T x \leq b_i, i = 1, 2, \ldots, m$$

- Objective function f is a convex quadratic function.

Optimization Problem: Why Useful

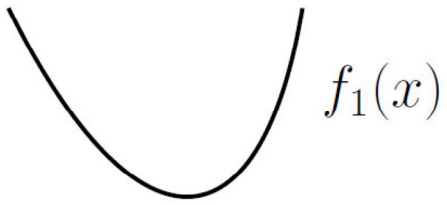
- Why formulate problems as optimization problems?
  - For many class of optimization problems, algorithms or algorithmic frameworks have been developed
  - Decouple “representation” and “problem solving”
- Lazy mode ☺
  - Formulate a problem as an optimization problem
  - Identify which class the formulation belongs to
  - Call the corresponding solver
  - Done!

Convex Optimization: Definition

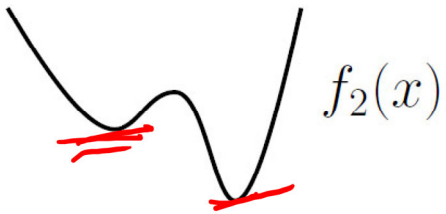
- Convex Optimization Problem:

$$\min_{x} f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

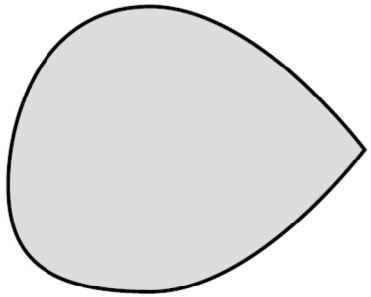
  - An optimization problem whose optimization objective f is a convex function and feasible region $\mathcal{F}$ is a convex set
  - A special class of optimization problem



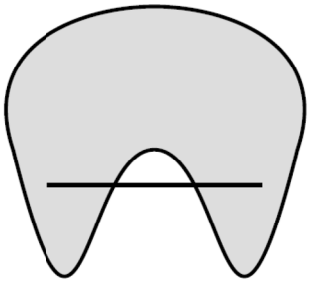
Convex function



Nonconvex function



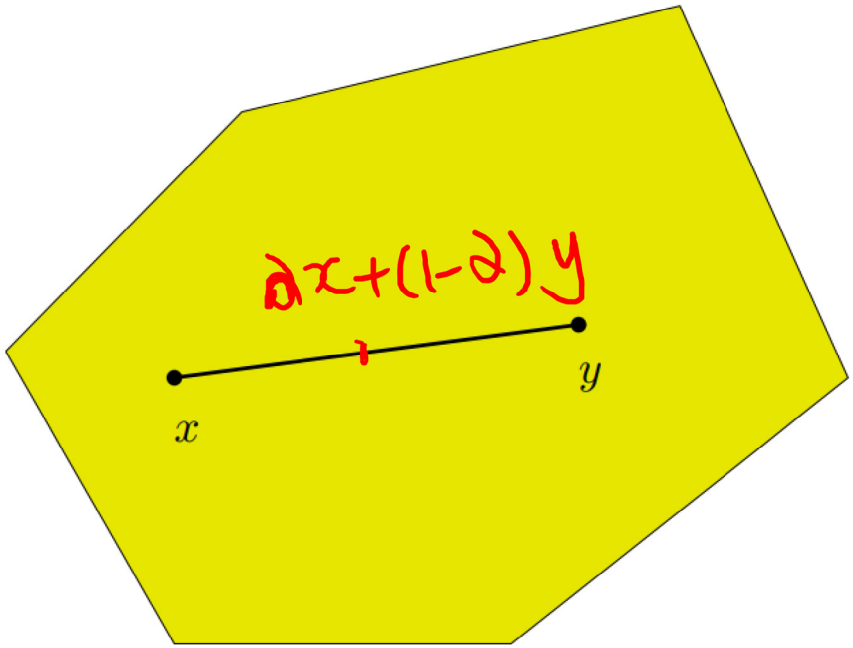
Convex set



Nonconvex set

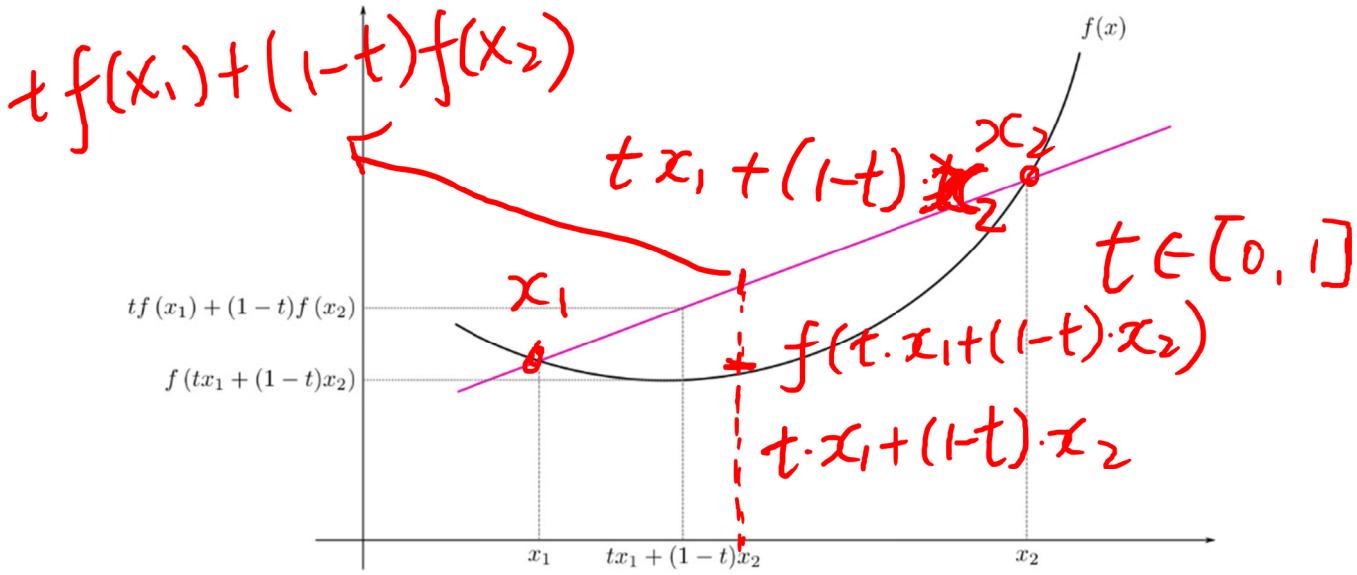
Convex Set

- A set $C \subseteq \mathbb{R}^n$ is convex if for $x, y \in C$ and any $\alpha \in [0, 1]$, we have
  - $\alpha x + (1 - \alpha) y \in C$



Convex Function

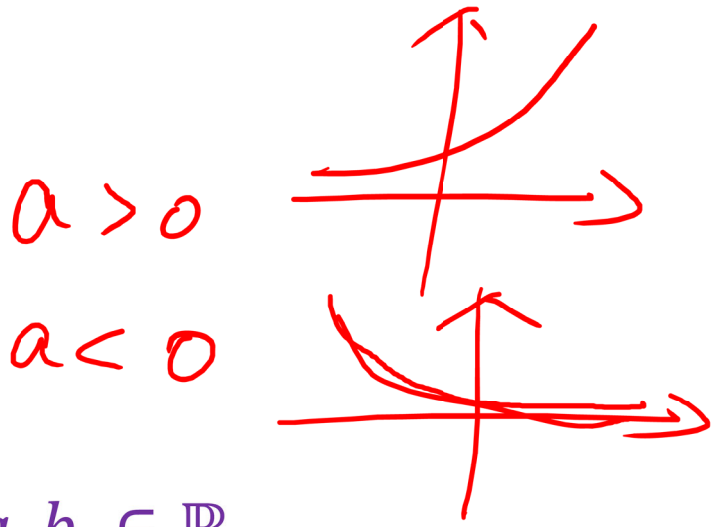
- Value in the middle point is lower than average value
  - Let $\mathcal{F}$ be a convex set. A function $f: \mathcal{F} \to \mathbb{R}$ is convex in $\mathcal{F}$ if $\forall x_1, x_2 \in \mathcal{F}, \forall t \in [0,1]$,
    $$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$$
  - If $\mathcal{F} = \mathbb{R}^n$, we simply say f is convex



Convex Optimization: Definition

- How to determine if a function is convex?
  - Prove by definition
  - Use properties
    - If f is a twice differentiable function of one variable, f is convex on an interval $[a, b] \subset \mathbb{R}$ iff (if and only if) its second derivative $f''(x) \geq 0$ in $[a, b]$
    - Sum of convex functions is convex
      - If $f(x) = \sum_i w_i f_i(x)$, $w_i \geq 0$, $f_i(x)$ convex, then $f(x)$ is convex
    - Convexity is preserved under a linear transformation
      - If $f(x) = g(Ax + b)$, g is convex, then $f(x)$ is convex

Examples on $\mathbb{R}$



- Convex
  - Affine: $ax + b$ on $\mathbb{R}$, for any $a, b \in \mathbb{R}$
  - Exponential: e^{ax}, for any $a \in \mathbb{R}$
  - Powers: x^a $(x > 0)$ for $\alpha \geq 1$ or $\alpha \leq 0$
  - Powers of absolute value: $|x|^p$ on $\mathbb{R}$, for $p \geq 1$
  - Negative entropy: $x \log x$ $(x > 0)$

regression

P-norm
$p = 2$ l-2 norm

classification

cross entropy

- Examples:
  - Is $f(x) = x^3, x \in \mathbb{R}$ a convex function?

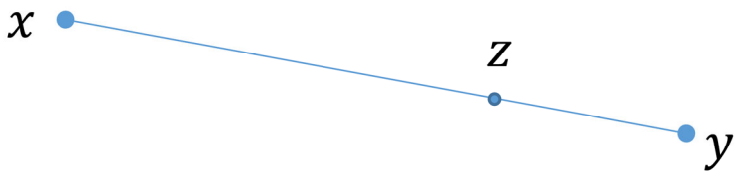
$f'(x) = 3x^2 \Rightarrow f''(x) = 6x$.

$x > 0$. $f''(x) > 0$.

$x < 0$. $f''(x) < 0$

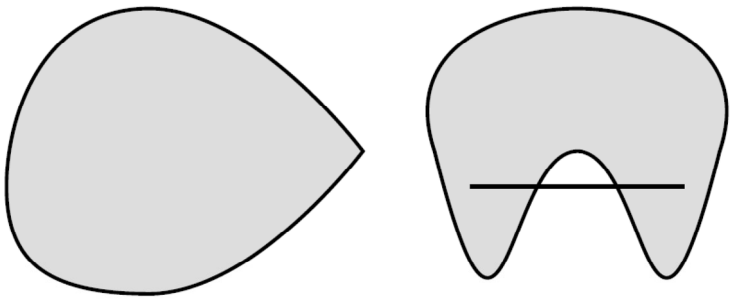
Convex Combination

- Convex combination
  - A point between two points
  - Given $x, y \in \mathbb{R}^n$, a *convex combination* of them is any point of the form $z = \theta x + (1-\theta)y$ where $\theta \in [0,1]$
  - When $\theta \in (0,1)$, z is called a strict convex combination of x, y



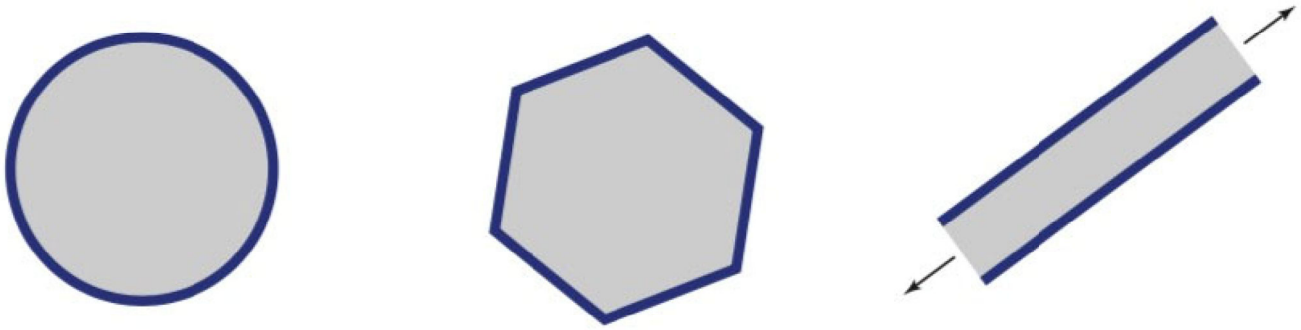
Convex Optimization: Definition

- Convex set
  - Any convex combination of two points in the set is also in the set
  - A set $\mathcal{F}$ is *convex* if $\forall x, y \in \mathcal{F}, \forall \theta \in [0,1]$,
$$z = \theta x + (1-\theta) y \in \mathcal{F}$$

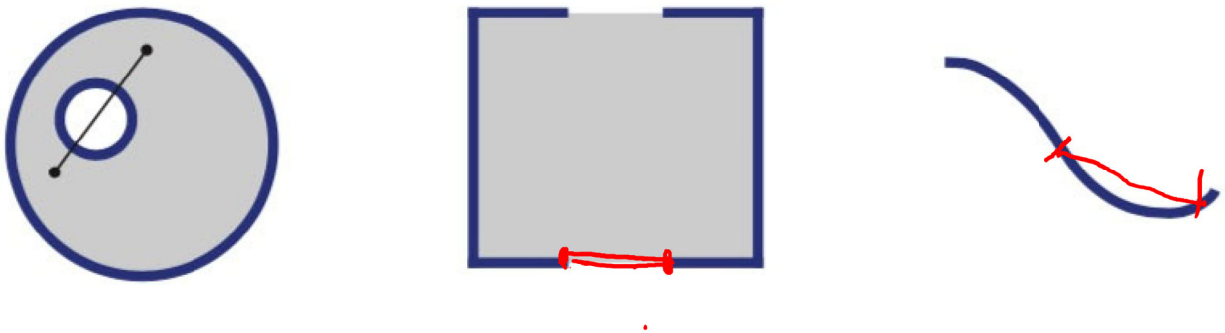


Convex & Nonconvex Sets

Convex:



Non-convex:



Concave function

- A function f is concave if $-f$ is convex
  - Let $\mathcal{F}$ be a convex set. A function $f: \mathcal{F} \to \mathbb{R}$ is *concave* in $\mathcal{F}$ if $\forall x, y \in \mathcal{F}, \forall \theta \in [0,1]$,
    - $f(\theta x + (1-\theta)y) \geq \theta f(x) + (1-\theta) f(y)$
  - Example
    - Affine: $ax + b$
    - Powers: x^a $(x > 0)$, $0 \leq \alpha \leq 1$
    - Logarithm: $\log x$ $(x > 0)$

$f''(x) \leq 0$.

- The following is a convex optimization problem if f is a concave function and $\mathcal{F}$ is a convex set

$$\max_x f(x)$$
$$\text{s.t. } x \in \mathcal{F}$$

minimize convex

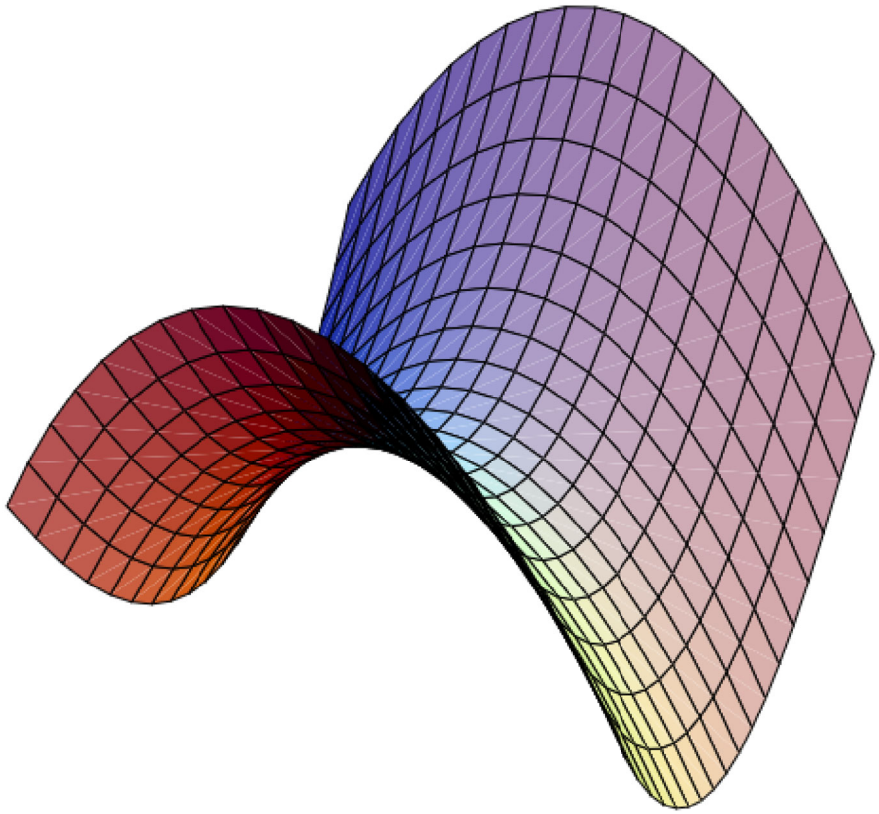
max concave

Affine Function

- An *affine function* is a function of the form $f(\boldsymbol{x}) = \boldsymbol{A}^T\boldsymbol{x} + b$ where $\boldsymbol{A} \in \mathbb{R}^n, b \in \mathbb{R}$
  - If a function f is both convex and concave in $\mathbb{R}^n$, then f is an affine function
  - An affine function is a linear function plus a translation/offset

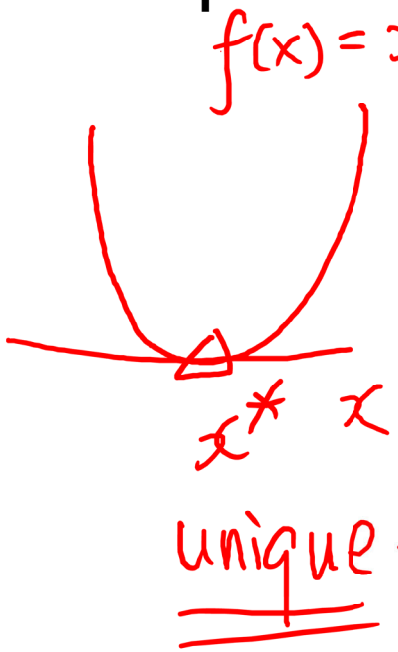
Can a Function be both Convex and Concave?

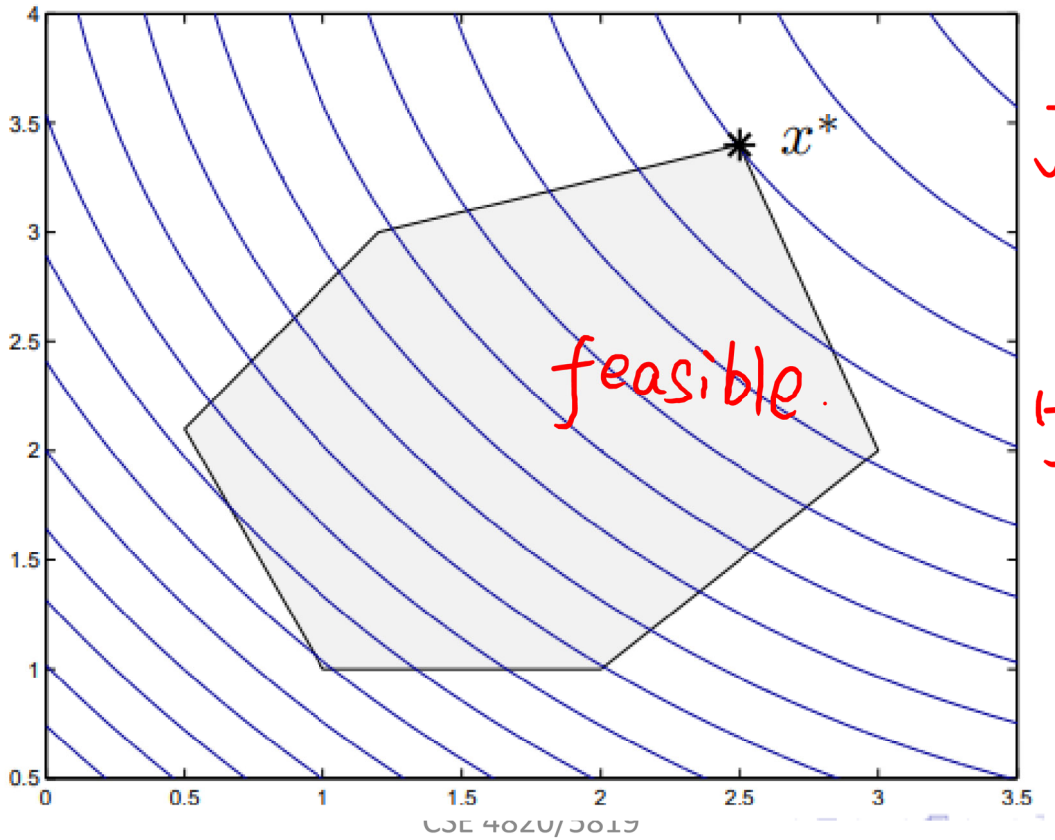
- Hyperbolic paraboloid



Convex Optimization: Local Optima=Global Optima

- If x^* is a local minimizer of a convex optimization problem, it is a global minimizer





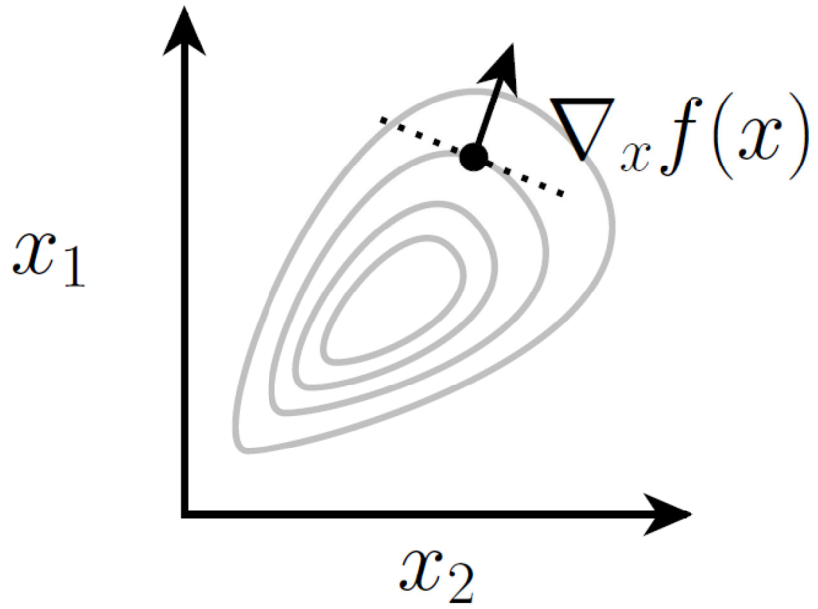
contours $f(x)$

if $f(x)$ is a convex function.

Convex Optimization: How to Solve

- For $f: \mathbb{R}^n \rightarrow \mathbb{R}$, *gradient* is the vector of partial derivatives
  - A multi-variable generalization of the derivative
  - Point in the direction of steepest increase in f

$$\nabla_x f(x) \in \mathbb{R}^n = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix}$$



Some Useful Derivatives

Linear Functions

$$f(x) = Ax \Rightarrow f'(x) = A$$

Example: $f(x) = 4x$

$f'(x) = 4$

Exponents

$$f(x) = Ax^n \Rightarrow f'(x) = nAx^{n-1}$$

Example: $f(x) = 3x^5$

$f'(x) = 15x^4$

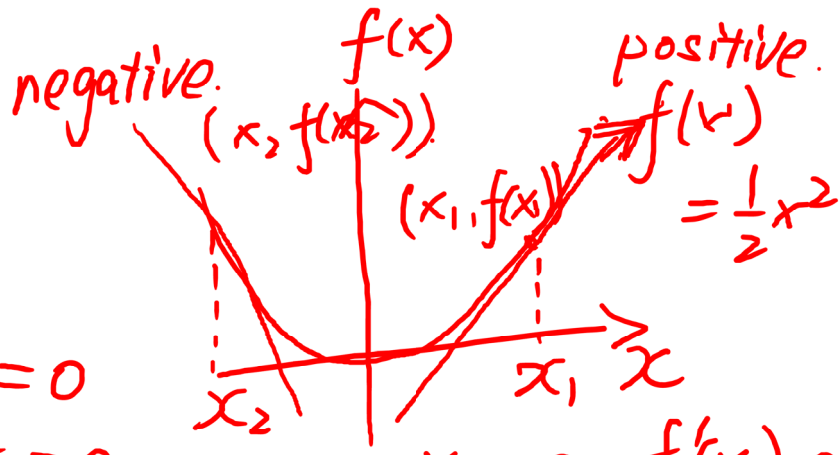
Logarithms

$$f(x) = A\ln(x) \Rightarrow f'(x) = \frac{A}{x}$$

Example: $f(x) = 12\ln(x)$

$f'(x) = \frac{12}{x}$

Gradient: Example



$n = 1$ $q = 0$

$P = 1$ $r = 0$

$x_1 > 0$ $f'(x_1) > 0$

$x_2 < 0$ $f'(x_2) < 0$

Examples

As a simple example consider the quadratic function $f : \mathbf{R}^n \to \mathbf{R}$,

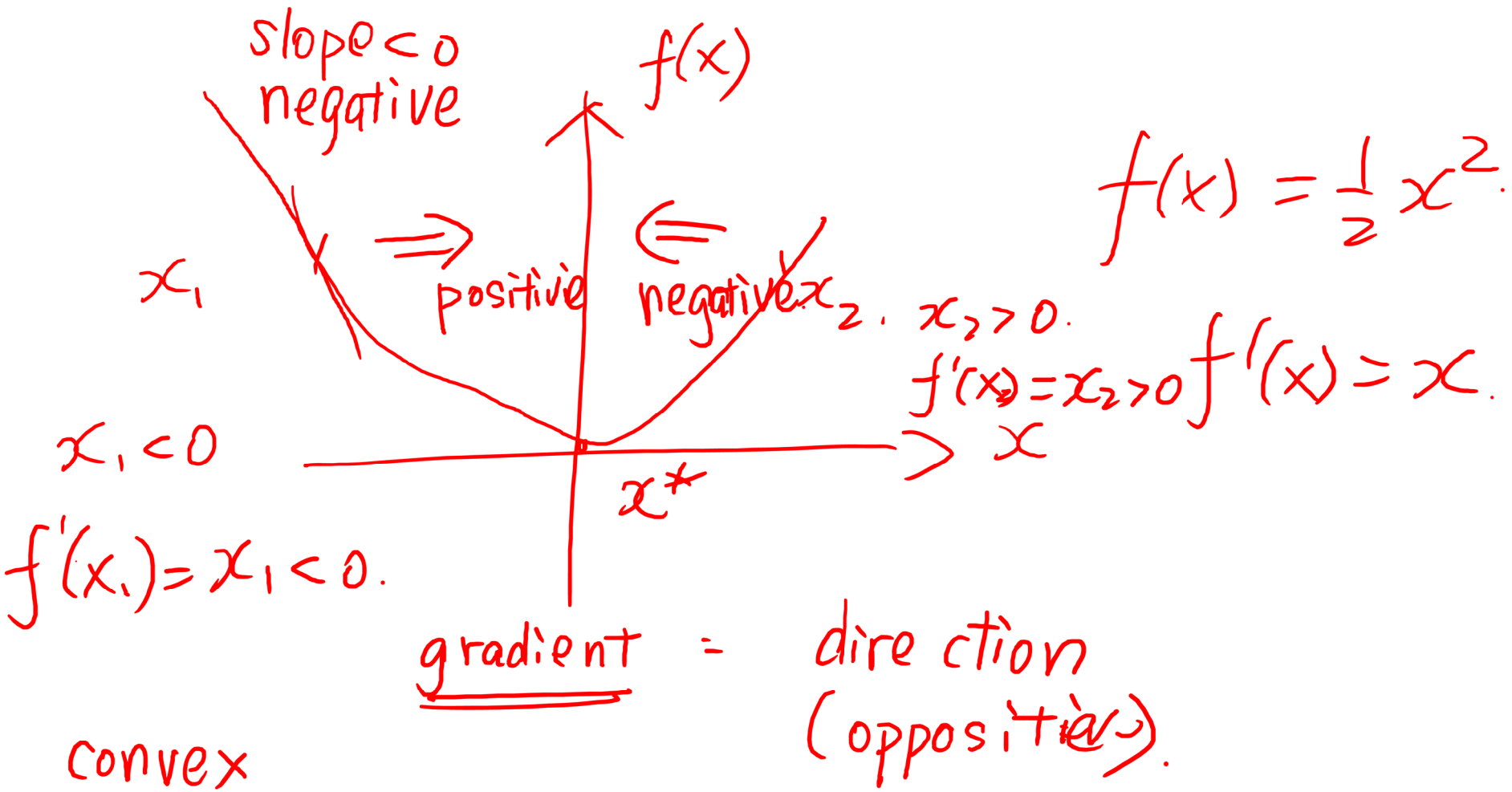
$$f(x) = (1/2)x^T Px + q^T x + r,$$

$f(x) = \frac{1}{2}x^2$

where $P \in \mathbf{S}^n$, $q \in \mathbf{R}^n$, and $r \in \mathbf{R}$. Its derivative at x is the row vector $Df(x) = x^T P + q^T$, and its gradient is

$$\nabla f(x) = Px + q.$$

$f'(x) = x$



x_1

x_2. $x_2 > 0$.

$f(x) = \frac{1}{2}x^2$.

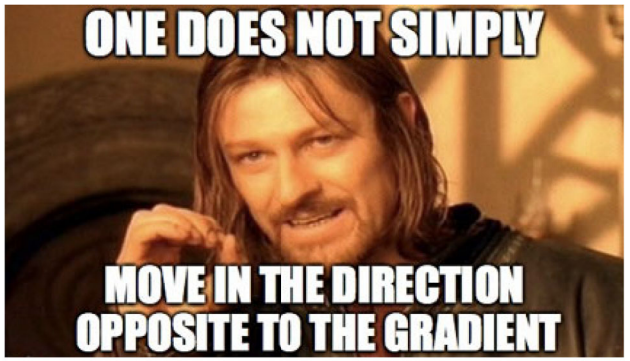
$f'(x) = x_2 > 0$ $f'(x) = x$.

$x_1 < 0$

$f'(x_1) = x_1 < 0$.

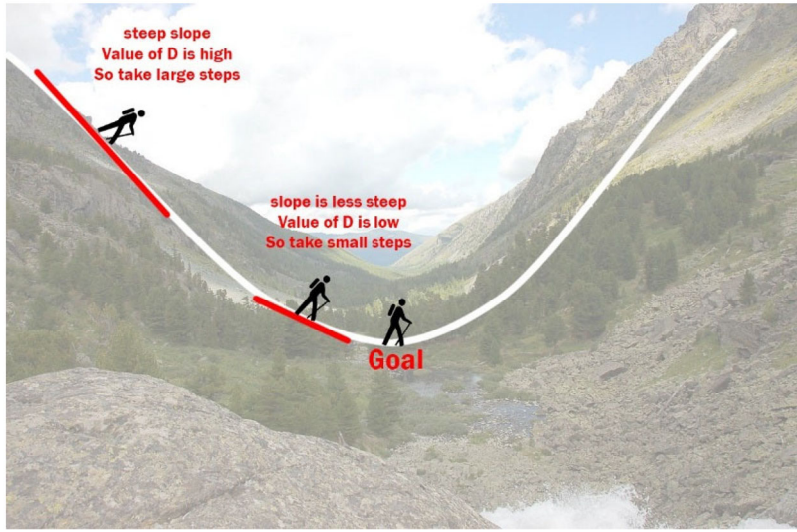
gradient = direction (opposite).

convex

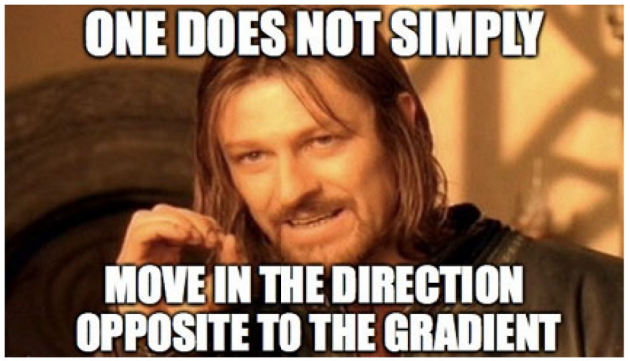


Gradient Descent

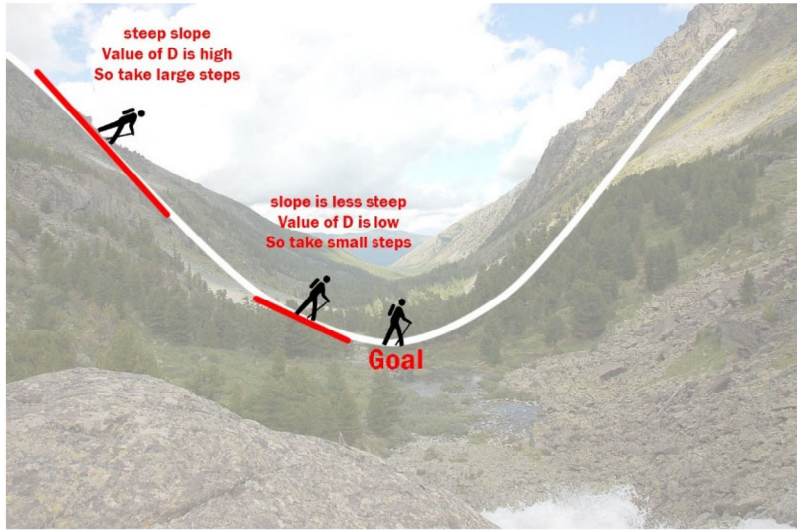
- *Gradient Descent*
  - A generic optimization algorithm capable of finding optimal solutions to a wide range of problems.
  - The general idea of Gradient Descent is to *tweak parameters iteratively in order to minimize a cost function.*



Gradient Descent



- It measures the local gradient of the error function with regard to the parameter vector θ, and it goes in the direction of descending gradient.
  - Once the gradient is zero, you have reached a minimum!



Basic Idea

1. **Initialize:** the weights θ randomly.
2. **Calculate the gradients:** G of objective function (cost/loss) w.r.t parameters.
   - This is done using partial differentiation: $G = \partial L(\theta)/\partial\theta.$
   - The value of the gradient *G* depends on the inputs, the current values of the model parameters, and the cost/loss function.
3. **Update the weights** by an amount proportional to G, *i.e.* $\theta = \theta - \alpha G$
4. Repeat until the objective function $L(\theta)$ stops reducing, or some other pre-defined **termination criteria** is met.

Convex Optimization: How to Solve

- Gradient descent: iteratively update the value of θ
- A simple algorithm for unconstrained optimization $\min_{\theta \in \mathbb{R}^n} f(\theta)$

Algorithm: Gradient Descent

Input: function f, initial point θ_0, step size $\alpha > 0$

Initialize $\theta \leftarrow \theta_0$

Repeat

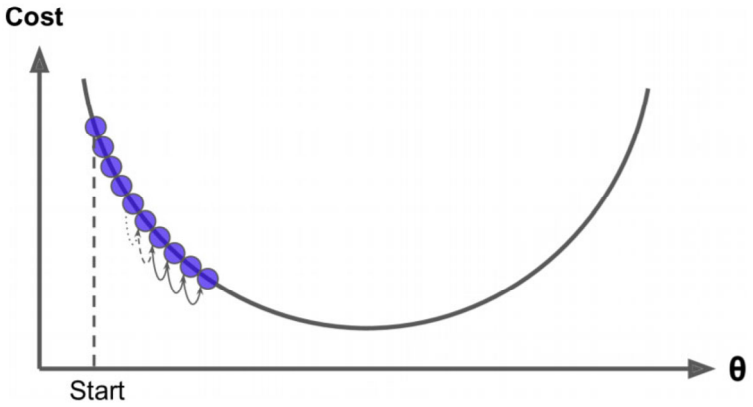
$$\theta \leftarrow \theta - \alpha \nabla_\theta f(\theta)$$

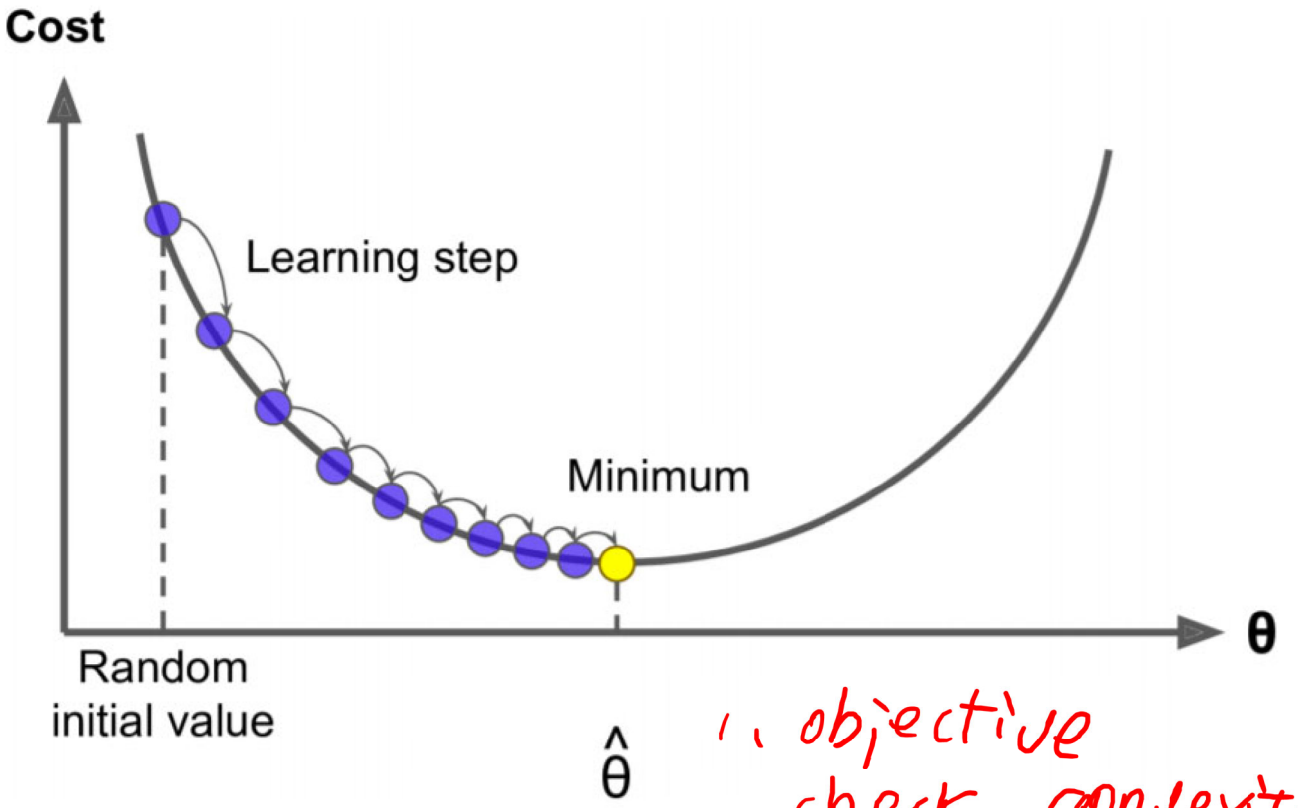
Until convergence

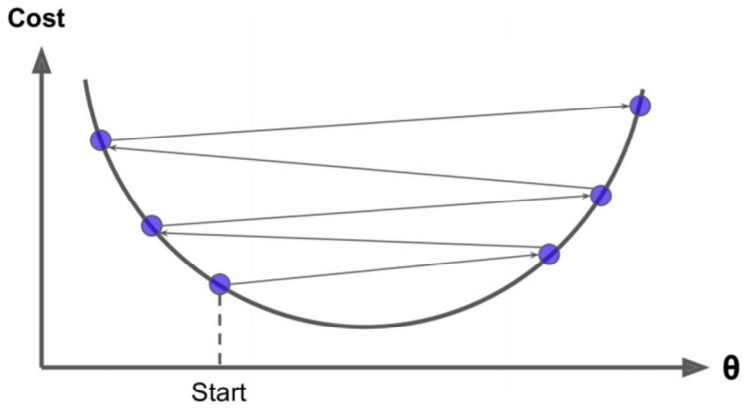
- Variants
  - How to choose θ_0, e.g., $\theta_0 = 0$
  - How to define "convergence", e.g., $\|\theta^{i+1} - \theta^i\|_2 \leq \epsilon$

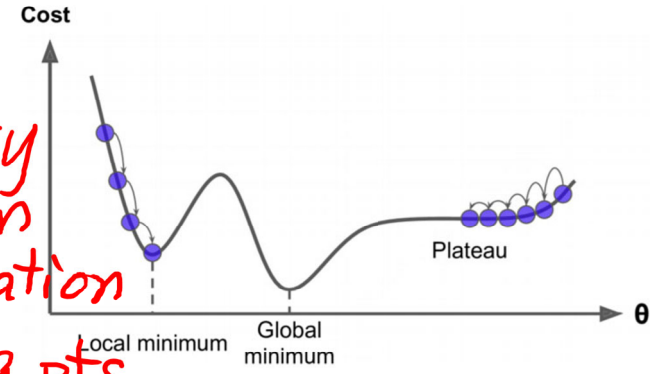
$\nabla_\theta f(\theta) \approx 0$

Gradient Descent









1. objective
check convexity
p-norm regression
cross-entropy classification

2. objective
~~non~~-convex
try multiple starting pts

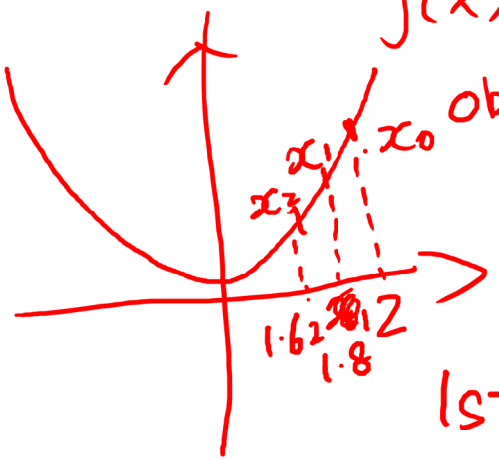
Quick Example

$f(x,y) = \frac{1}{2}x^2 + \frac{1}{2}y^2$

$\nabla = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$

$f(x) = \frac{1}{2}x^2 \quad f'(x) = x.$

objective $x_0 = 2 \quad \alpha = 0.1$



$f'(x_0) = 2$

1st iteration $\quad \theta \leftarrow \theta - \alpha \cdot \nabla f(x)$

$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} - \alpha \cdot \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$

$x_1 = x_0 - 0.1 \times x_0$

$= 2 - 0.2 = 1.8$

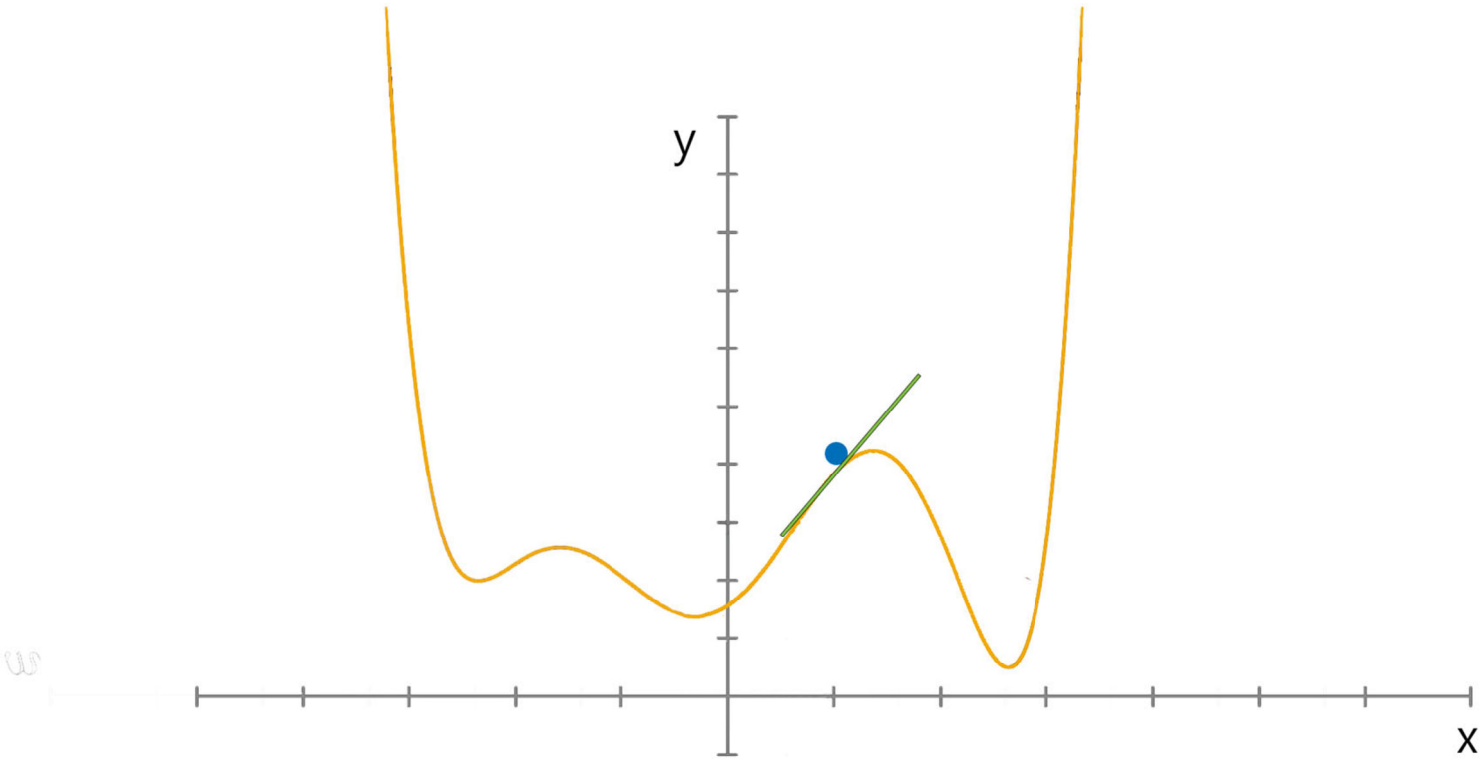
2nd iteration $\quad f'(x_1) = 1.8.$

$x_2 = x_1 - 0.1 \times x_1$

$= 1.8 - 0.1 \times 1.8$

$= 1.62$

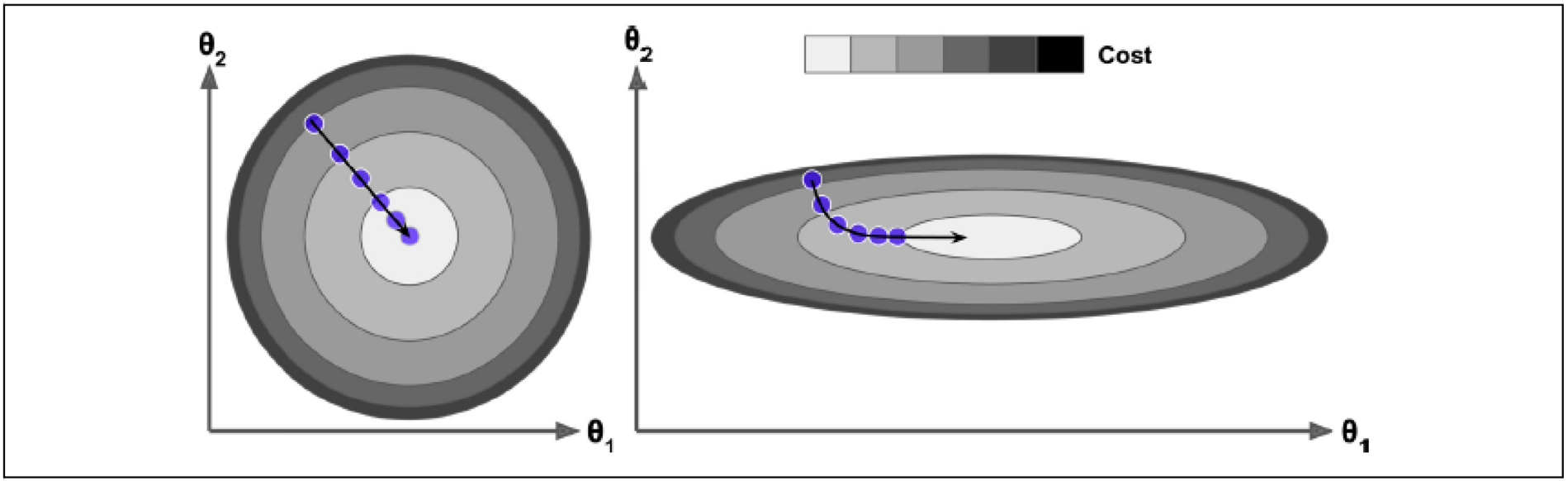
Gradient Descent



Feature Scaling

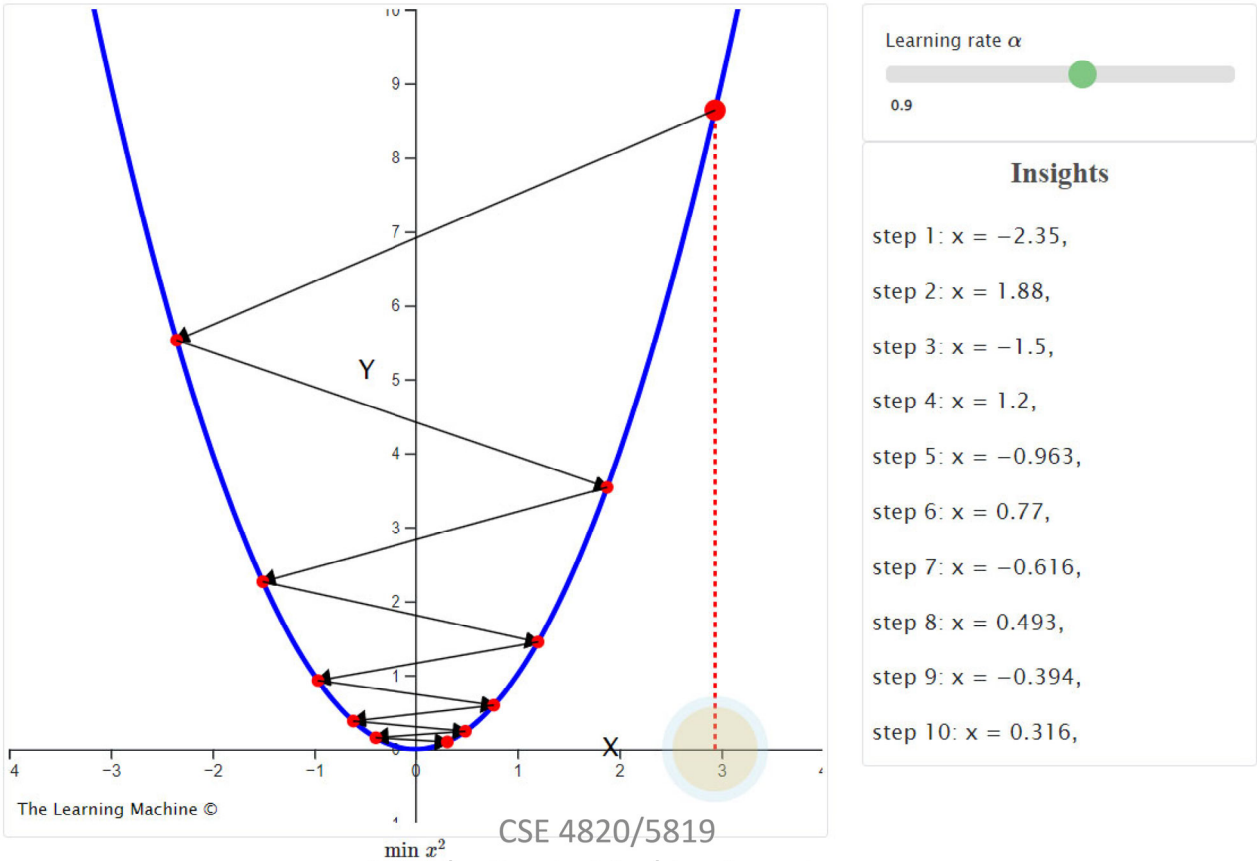
Standardization
$\theta_1, \theta_2 \rightarrow [0, 1]$
$[-1, 1]$
min-max normalization
z-score standardization

- When using Gradient Descent, you should ensure that all features have a similar scale (e.g., using Scikit-Learn's StandardScaler class), or else it will take much longer to converge.

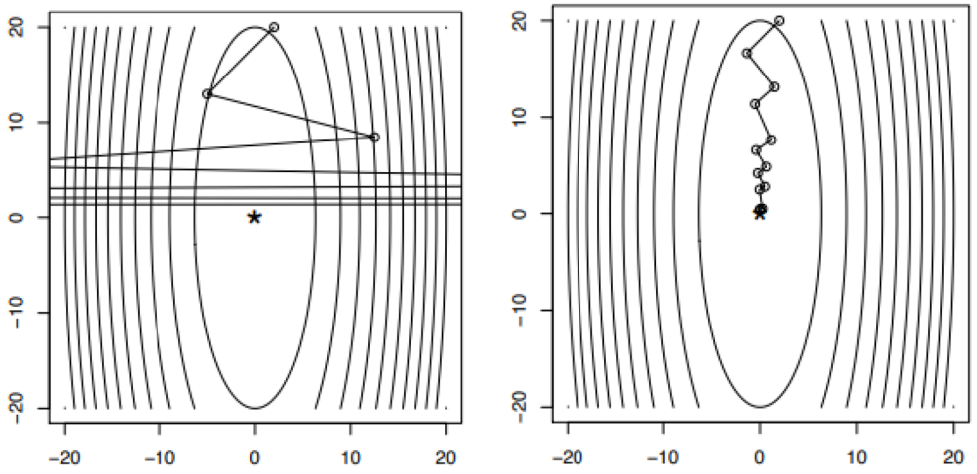


Gradient Descent: Interactive Demo

- https://the-learning-machine.com/article/optimization/gradient-descent



Step Size

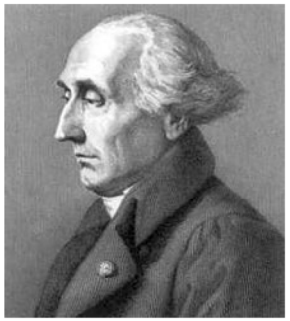


- Fixed step size
  - Keep α the same all the time
- Backtracking line search
  - Fix a parameter $0 < \beta < 1$, then at each iteration, start with $\alpha = 1$, and while

    $$f(\theta) - f\left(\theta - \nabla_\theta f(\theta)\right) \leq \frac{\alpha}{2} \|\nabla_\theta f(\theta)\|^2,$$

    Update $\alpha = \beta\alpha$
    - Shrinking stops if the objective function can decrease as expected

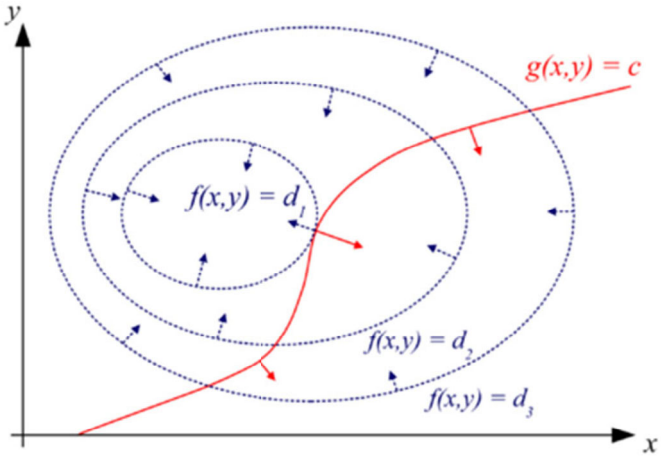
Using Lagrange Multiplier



Joseph Lagrange

- Given an optimization problem
  - Equality constraints

$$\min_x f(x)$$
$$\text{s.t. } h_j(x) = 0$$



- Lagrangian function form:

$$L(x, \lambda) = f(x) + \sum_j \lambda_j \, h_j(x)$$

- Stationary point conditions for equality constraint:

$$\frac{\partial L}{\partial x_i} = \frac{\partial f}{\partial x_i} + \sum_j \lambda_j \frac{\partial h_j}{\partial x_i} = 0 \qquad \frac{\partial L}{\partial \lambda_j} = h_j = 0$$

Example

- Quadratic objective and equality constraint

$$\min_{x} \quad f(x) = x_1^2 + 10x_2^2$$
$$\text{s.t. } 100 - \left(x_1^2 + x_2^2\right) = 0$$

- Lagrangian
- Stationary point conditions
- Stationary points

• Quadratic objective and equality constraint

$$\min_x \quad f(x) = x_1^2 + 10x_2^2$$

$$\text{s.t. } 100 - (x_1^2 + x_2^2) = 0$$

$$L(x_1, x_2, \lambda) = f(x) + \lambda\left(100 - (x_1^2 + x_2^2)\right)$$

$$= x_1^2 + 10x_2^2 + \lambda(100 - (x_1^2 + x_2^2))$$

$$\frac{\partial L}{\partial x_1} = 2x_1 + 0 + 0 - 2\lambda x_1 + 0$$

$$= 2x_1 - 2\lambda x_1 = 0 \Rightarrow (1-\lambda)\cdot x_1 = 0$$

① $x_1 = 0$. $x_2 = \pm 10$ $\lambda = 10$. $(0, 10)$ $(0, -10)$

$$\frac{\partial L}{\partial x_2} = 0 + 20x_2 + 0 + 0 - 2\lambda x_2$$

$$= (10 - \lambda)\cdot x_2 = 0$$

$$\frac{\partial L}{\partial \lambda} = 100 - (x_1^2 + x_2^2) = 0$$

② $x_2 = 0$. $x_1 = \pm 10$ $\lambda = 1$. $(10, 0)$ $(-10, 0)$

Pick the (x_1, x_2) minimize $f(x)$

- Quadratic objective and equality constraint

$$\min_{x} \quad f(x) = x_1^2 + 10x_2^2$$
$$\text{s.t. } 100 - \left(x_1^2 + x_2^2\right) = 0$$

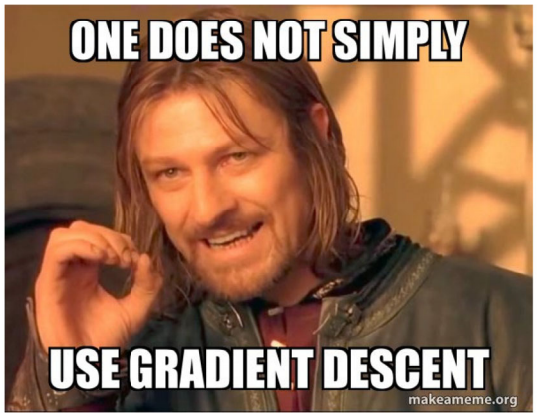
Convex Optimization: How to Apply

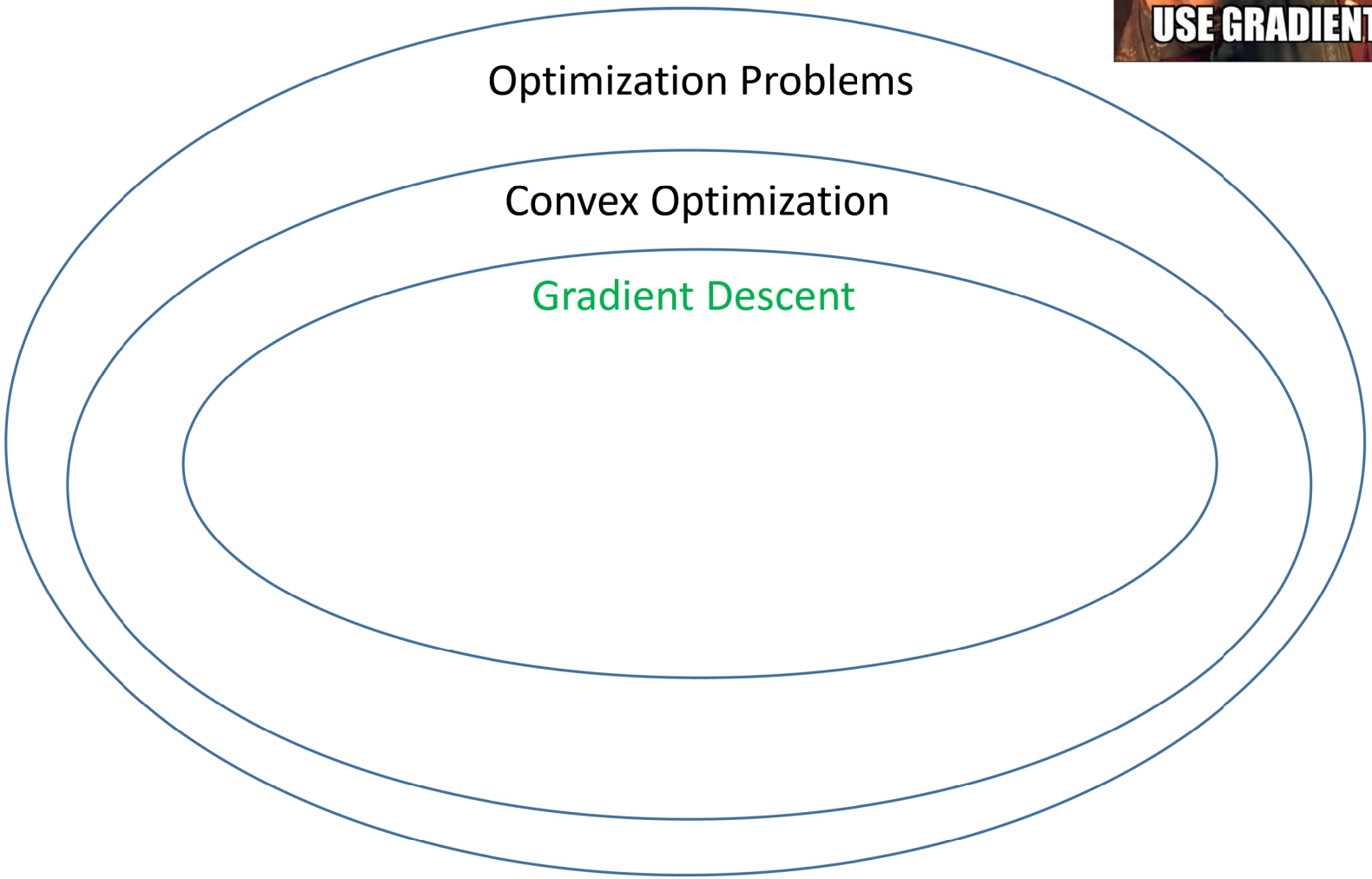
- Model a problem as a convex optimization problem
  - Define variable, feasible set, objective function
  - Prove it is convex (convex function + convex set)
- Solve the convex optimization problem
  - Build up the model
  - Call a solver
    - Examples: cvxpy (Python), fmincon (MATLAB), cvxopt (Python), cvx (MATLAB)
- Map the solution back to the original problem

Simple Demo

- Google Colab Demo
  - cvxpy

Summary


ONE DOES NOT SIMPLY
USE GRADIENT DESCENT
makeameme.org


Optimization Problems
Convex Optimization
Gradient Descent

Reading Materials

- Hands-on Machine Learning
  - Chapter 4: Gradient Descent
- Convex Optimization
  - Chapter 2 Convex Sets (2.1, 2.2)
  - Chapter 3 Convex Functions (3.1)
  - Chapter 4 Convex Optimization Problem (4.1, 4.2)

Convex Optimization: Additional Resources

- Text book
  - *Convex Optimization, Chapters 1-4*
    Stephen Boyd and Lieven Vandenberghe
    Cambridge University Press
    https://web.stanford.edu/~boyd/cvxbook/
- Online course
  - Stanford University, Convex Optimization I (EE 364A), taught by Stephen Boyd
    - http://ee364a.stanford.edu/courseinfo.html
    - https://youtu.be/McLq1hEq3UY