Fundamentos de optimización:

Hoy hablaremos de conceptos básicos en OPTIMIZACIÓN
Hay dos grandes familias, CONTINUA y DISCRETA.

(1) Qué es la optimización?

Def: Un problema de optimización consiste de tres partes:

(1) Un conjunto finito X llamado el conjunto factible (X representa las posibilidades que tenemos)

(2) Una función $f : X \to \mathbb{R}$ llamada la función objetivo ($f(x)$ representa nuestra "valoración" del item x)

(3) Un elemento de $\{MAX, MIN\}$ (que nos aclara si queremos maximizar ó minimizar nuestra "valoración")

Def: "Resolver un problema de optimización" tiene dos significados posibles

Encontrar el VALOR ÓPTIMO

$$\alpha^* = \min \{ f(x) : x \in X \}$$

α^* es un número real

Encontrar un elemento $x^* \in X$ con $f(x^*) = \alpha^*$.

x^* es UN ÓPTIMO del problema.

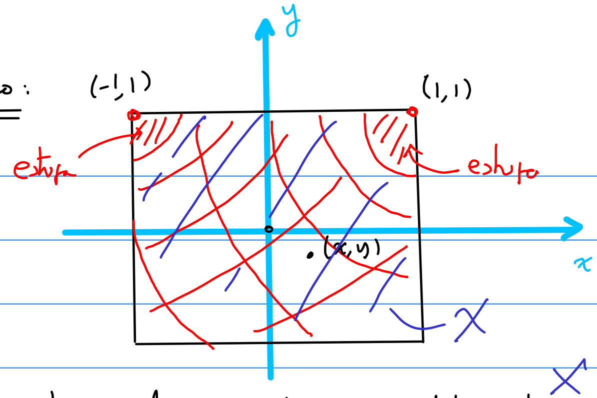
x^* es un elemento de X (típicamente un VECTOR)

$\rightarrow$ Este es más difícil.

Obs: Un problema de optimización puede tener varios óptimos x^* así que definimos el CONJUNTO DE MINIMIZADORES

$$\operatorname{argmin} = \{ x \in X : f(x) = \alpha^* \}$$

Ejemplo:



La temperatura de un punto (x,y) del cuarto X está dada por

$$f(x,y) = \frac{1}{\|(x,y)-(1,1)\|+1} + \frac{1}{\|(x,y)-(-1,1)\|+1}$$

PROBLEMA: (a) Encuentre la temperatura mínima que ocurre en el cuarto

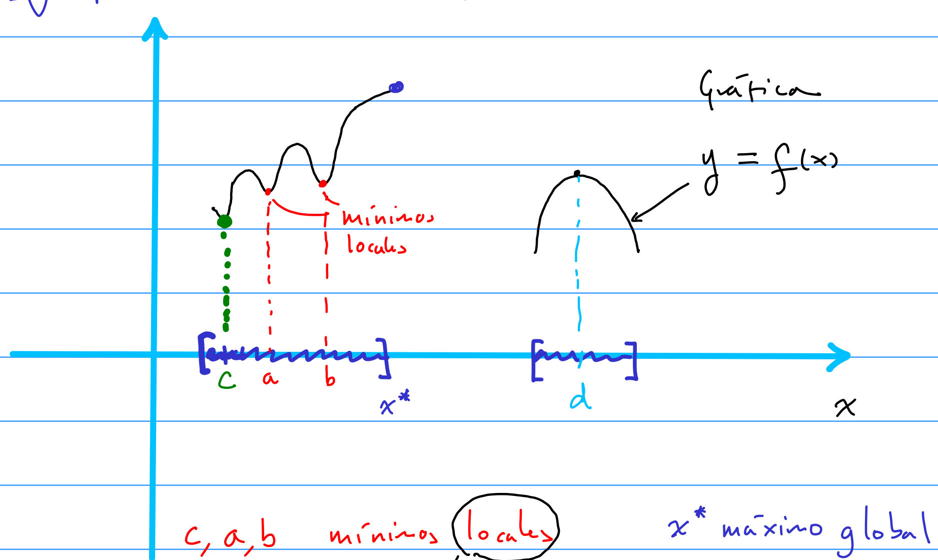
(b) Encuentre los lugares donde ese mínimo se alcanza.

(a) $t^* = \min \{ f(x,y) : (x,y) \in X \}$

(b) $\operatorname{argmin} \{ f(x,y) : (x,y) \in X \}$.

Este es un problema de optimización continua porque nuestro conjunto de opciones es infinito y los valores posibles de (x,y) que las parametrizan varían continuamente. La función objetivo f es diferenciable y por ello el problema puede atacarse con los métodos del Cálculo (multiplicadores de Lagrange).

Ejemplo: (LOCAL vs Global).



c, a, b mínimos locales $\qquad$ x^* máximo global

c mínimo global $\qquad$ x^*, d máximo local

Para todo z cerca de a
$f(z) \geq f(a)$ $\Big]$ — a es mínimo local

Nosotros nos concentraremos en problemas de optimización DISCRETA, es decir aquellos en los que X es un conjunto finito.
La dificultad es que X típicamente va a ser MUY grande, tanto que enumerarlo exhaustivamente, para buscar el mínimo valor mediante fuerza bruta (evaluando $f(y)$ para todo y en X recordando el más pequeño) es imposible. ¿Qué hacer entonces?...

OPTIMIZACIÓN DISCRETA ó COMBINATORIA.

Estos problemas son típicamente mucho más diversos que sus análogos continuos, cada problema tiene sus propias técnicas

PROBLEMA FÁCIL

Ejemplo: Sea G un grafo dirigido con pesos $\ell \geq 0$ en las aristas y sea $s \in V(G)$ un vértice fijo.

$$d_\ell(s,x) = \min \left\{ \ell(P) : \begin{array}{l} P \text{ es un camino} \\ \text{dirigido desde } s \text{ hasta } x \end{array} \right\}$$

con $P = (s, x_1, .., x_{k-1}, x)$ $\quad \ell(P) = \ell(s,x_1) + \ell(x_1,x_2) + .. + \ell(x_{k-1}, x)$

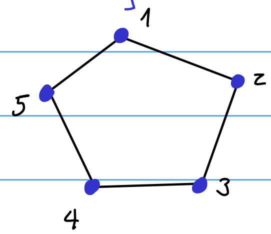
Hay un número exponencial de caminos de s a x
pero el algoritmo de Dijkstra nos da solución $O((m+n)\log(n))$

PROBLEMA EJEMPLO: *

(Subconjuntos independientes de máximo peso).

Def: Sea G un grafo no dirigido finito.
Un conjunto $S \subseteq V(G)$ es <u>independiente</u> ssi
$\forall a,b \in S \; [(a,b) \notin E(G)]$

Por ejemplo si G :



$\{1,3\}$ es independiente

$\{2,4,5\}$ **no** es independiente porque $(4,5) \in E(G)$

Aplicación: Sea G el grafo con
$V(G)$ — "Materias ofrecidas el próximo semestre que quiero tomar"
$(a,b) \in E(G) \iff$ a y b tienen horarios en conflicto

PREGUNTAS: (1) ¿Cuál es el número máximo de materias que puedo tomar (sin conflicto de horarios)

(2) ¿Cuál es el número máximo de créditos que puedo tomar?

RESPUESTAS: (1) Busque el conjunto indep de máximo cardinal (2 en G del dibujo)

(2) Deberíamos hacer que no todos los vértices valgan igual poniéndoles pesos (el número de créditos de cada materia)

Def: Una función de pesos en los vértices w para un grafo G es $w: V(G) \longrightarrow \mathbb{R}_{\geq 0}$

Con estos preliminares podemos definir el problema Weighted Independent Set (WIS) así:

Def: [WIS problem]

Dado un grafo no dirigido G con pesos ≥ 0 en los vértices w, definimos el

$$\bar{w}(S) = \sum_{e \in S} w(e)$$

Peso de un subconjunto S

y definimos el problema de OPTIMIZACIÓN DISCRETA

$$WIS(G) = \max \{ \bar{w}(S) : \underset{S \text{ independiente}}{S \subseteq V(G)} \}.$$

Siendo precisos, este problema tiene

(1) Conjunto factible $X = \{ S \subseteq V(G) : S \text{ es independiente} \}$

(2) Función objetivo $\bar{w} : X \to \mathbb{R}$ $\quad \bar{w}(S) = \sum_{s \in S} w(s)$

(3) Queremos MAXIMIZAR

INSTANCIA:

V	1	2	3	4
w	1	4	5	4

(1) Cuántos subconjuntos independientes hay? Cuántos si hay n vértices?

(2) Cuánto vale $WIS(G)$?

(3) Encuentre $S^* \in \underset{S \in X}{\operatorname{argmax}} \; \bar{w}$

En general? próxima clase.

Obs: No conocemos un algoritmo "rápido" para calcular $WIS(G)$ para grafos generales (pese a MUUCHOS intentos) y se cree que no puede existir [Has oído de P vs NP? INVESTIGAR en Google...]