Wed, March 22, 2023
Lecture #25
MSSC 6000

Announcements

* Normal Office Hours today, 2:30 - 3:30 on Teams and in-person CU 307
* HW 4 will be assigned on Friday, due the following Friday (1 problem)

Topic 10 - Introduction to Metaheuristics

In the 1st half of the semester, we focused on ways to find optimal solutions.

Downside: - difficult to design algo
- very slow (even though very slow)

Ex: Travelling Salesman still takes exponential time $O(n^2 \cdot 2^n)$ which is too slow to be useful.

Metaheuristics:

- General problem solving paradigms that can easily adapted to many problems
- They for good solutions, and rarely find actually optimal ones
- Pretty fast

Similar setup:

* Search space of candidates
* Every candidate has a score ("fitness" or "quality")
* Could have constraints
* Goal: Find a candidate with a good score that satisfies the constraints

[in the abstract we'll talk about maximizing, but sometimes you want to minimize]

Our problems will be categorized as:

discrete or continuous

↑ finite search space

↑ infinite search space

Demos: "matplotlib" pip install

(1) TSP random and random greedy

Best ≈ 20.95 (random)

7.12 (random greedy)

(2) Steepest Ascent Hill-Climbing

≈ 6.487

(3) Simulated Annealing

≈ 6.300

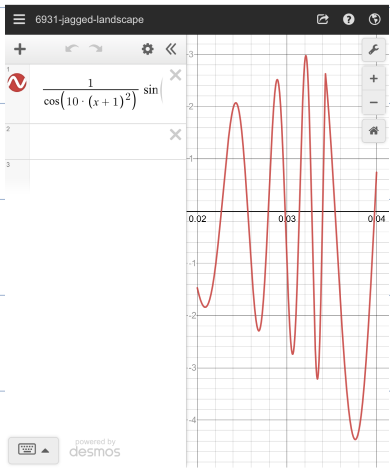
$\underbrace{\hphantom{xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx}}$

a discrete problem

Ex: Find the maximum value of

$$f(x) = \frac{1}{\cos(10(x+1)^2)} \cdot \sin\left(\min\left((x+1)^{100}, \frac{1}{x}\right)\right)$$

on the interval $0.02 \le x \le 0.04$.



This one you could maybe do with calculus, but the "min" makes that hard, and most interesting problems are:

- are in many more variables
- implicit (solution of some complicated diff. eq.)

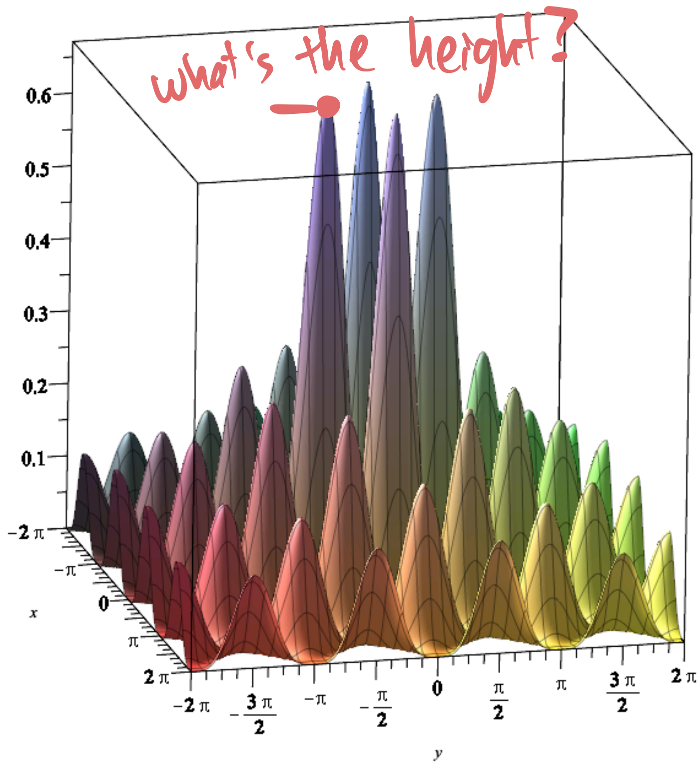
Most of the search spaces we'll look at are not 1 dimensional

<u>Ex:</u> TSP – finite, 0 dimensional, but big

Helpful to think about "landscape pictures"

~ metaphor or mental picture for a search space

Ex: $\dfrac{\sin^2(x-y)\cdot\sin^2(x+y)}{\sqrt{x^2+y^2}}$



Goal: Find the top of the tallest hill.

Gradient Ascent/Descent

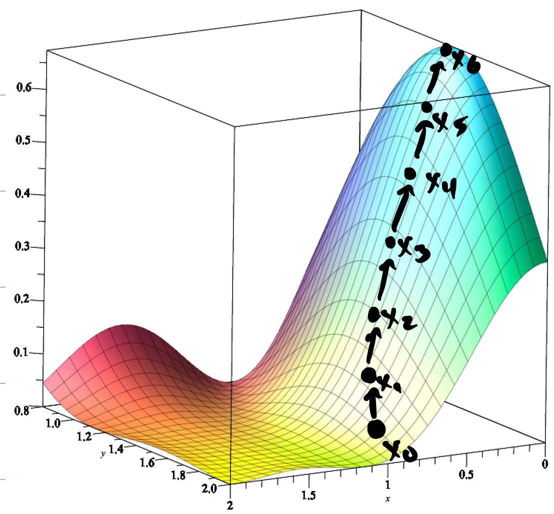
* Optimization method you learn in some math classes
* If your function $f(x,y)$ is differentiable, then at any (x,y) point you can compute the "gradient" of f at that point.
* The gradient is a vector that

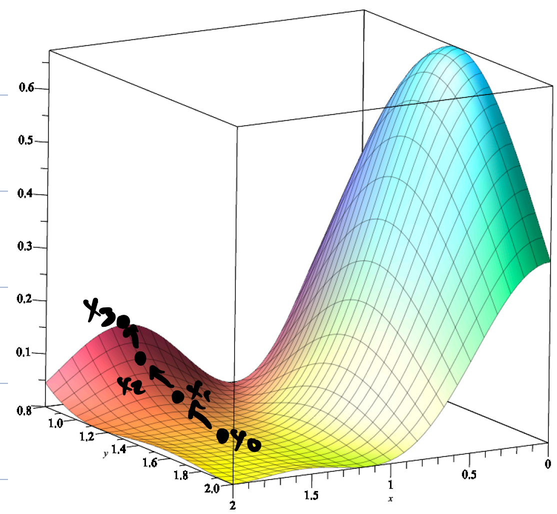
always points in the direction of steepest ascent.

Process:

(1) Start at a point
(2) compute the gradient
(3) move a little in that direction
(4) repeat

Where do you end up? Depends on where you start.





Requires calculus ($f(x,y)$ to be differentiable)