Lecture 9: How can computation... pick best data values for goal?

Data is Important

Every field values their data
- Science, engineering, medical, business, sports
- Data source: Measurements, simulations, mathematical functions

What does one often want to do with data?
- Pick set of parameters that lead to best result
  - Which materials at which temps are strongest?
  - Which medical procedure saves most lives at least cost?
  - What stock portfolio gives best profits while minimizing risk?
  - Which pitchers most likely to get next batters out?

General Question: Which parameter values optimize desired function while meeting some constraints?

Data Example: Electroencephalography (EEG)

Assume two inputs: Time and frequency; Assume Output: Spectral signal
How to find (x, y) that maximizes z?

Data Example: Quantum Hall effect

2D electrons: Topographic map of random potential by SPM microscopy
How to find (x, y) that maximizes z?
Much simplified Scenario: Business Owner

Consider single input variable:
- Unit Price

Imagine: Profit = f(unit price)
- Magically know f()
- f() might be very complex...

How would you set the price to maximize profit?

How can you use computation to find the optimal price?

Naive Optimization Approach

Evaluate set of options and pick best
- Pick some prices (x values) (every 10¢ thru $11)
- Compute profit = f(price) and store in profit List
- Naïve approach: Exhaustive search through entire list for maximum
  - Code in Lecture 7

Why won’t this approach always work?

Why doesn’t Naïve approach always work?

What if too many data points to test all?
- MB (10^6), GB (10^9), TB (10^12) of data on disk

What if very costly to compute f(x)?
- Computing f(x) could be running a day-long simulation on 1000 processors

Can’t explore entire data set
- Must explore only small number of points

What would you do?

Better Optimization Approach

Pick 1st random point
Pick 2nd point in dir A
Worse? Switch dirs
Pick 3rd point in dir B
Better? Keep going
Pick 4th point in dir B
Better? Keep going
Pick 5th point in dir B
- Worse? Switch back...
  - Settle on 6th point...

Official name: gradient ascent (descent)
What Can Go Wrong?

Can find local maxima instead of global!

How could you improve algorithm?

Implementing Optimization in Scratch

Scenario: You own a business upholstering cushions

Customer wants you to cover a bolster (cylindrical cushion) for $30
  - Must contain 10m³ of stuffing
  - Don’t care about the dimensions (radius or height)

Your costs include material
  - Less material you use, the more profit you’ll make...

What Improvements Could You Make?

Try multiple times
  - Different random starts
  - Keep best result

“Simulated annealing”
  - Sometimes continue when see worse results
  - Pick worse point with some probability p
  - Gradually decrease p over time

How to Solve Bolster Problem?

Useful formulas:

\[ V = \pi \cdot r^2 \cdot h \]
\[ SA = 2\pi r^2 + 2\pi rh \]

Which are known? (or fixed? or constrained?)

\[ V = 10 \text{ m}^3 \text{ of stuffing} \]
Which are you trying to optimize?

\( SA \) - minimize amount of material needed

How can you solve?

Want to vary only one variable: Specify \( h \) in terms of \( r \) (given fixed \( V \))
Try different values for radius; See which gives minimal \( SA \)

\[ h = \frac{V}{\pi r^2} \]
\[ SA = 2\pi r^2 + \frac{2V}{r} \]
What does function look like?

How should you search for radius leading to minimal Surface Area?

Questions to Ask and Answer

What variable are we varying?
- Radius

What variable are we trying to minimize?
- Surface area

How do we know the value of the surface area?
- \[ SA = 2\pi r^2 + \frac{2V}{r} \]

How should we initialize radius?
- Don’t know
- Ask the user!

How much should we increase it on each iteration?
- Don’t know, so ask the user!

How do we know we’ve found the minimum?
- \( SA_{\text{new}} > SA_{\text{prev}} \)
- Previous radius is the minimum

Simple Implementation

Keep trying larger values of radius until \( SA > \text{prev} \ SA \)

For \( V = 10 \text{m}^3 \)
- Best radius = 1.2m
- Leads to \( SA = 25.71 \text{m}^2 \)
- Height = 2.21m

Code structure similar for any optimization problem
- Replace \( f(r) \)

Today’s Summary

Optimization
- How to use computation find parameters leading to best (max, or min) result
- Simplest: Search through parameter space linearly, stop at max/min; more sophisticated techniques to find global optimum
- Many engineers and scientists use computation for optimization

Reading:
- Ch 3: Numerical Computation and a Study of Functions from “Great Ideas in Computer Science”, pages 80-103

Announcements
- Homework 3 Due Wednesday
- Programming Project 1 Available soon: Due 2+ weeks
  - Create music video or animate poem