Beyond binary search

Prof. Ramin Zabih

http://cs100r.cs.cornell.edu

Cornell University
Computer Science

Administrivia

- Assignment 4 is due tomorrow
- Prelim schedule
  - P2: Thursday Nov 1
    - Or possibly Tuesday Nov 6
  - P3: Thursday Nov 29

Beyond binary search

- Can we find the minimum of a 1D convex function without evaluating derivatives?
  - We'll use a similar interval-shrinking method
  - Intervals will be slightly more complex
- Basic idea:
  - Current interval is [a, b]
    - Assume this is valid (minimum lies inside)
  - We know the values of f(a) and f(b)
  - Compute f(c) and f(d), where c and d lie inside the interval [a,b]
  - Create a smaller interval

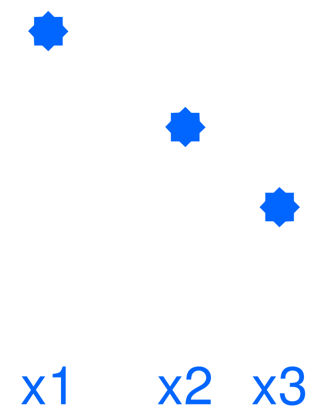
Key insight

- We will take advantage of this fact:
- Suppose:

  [A] x1 < x2 < x3

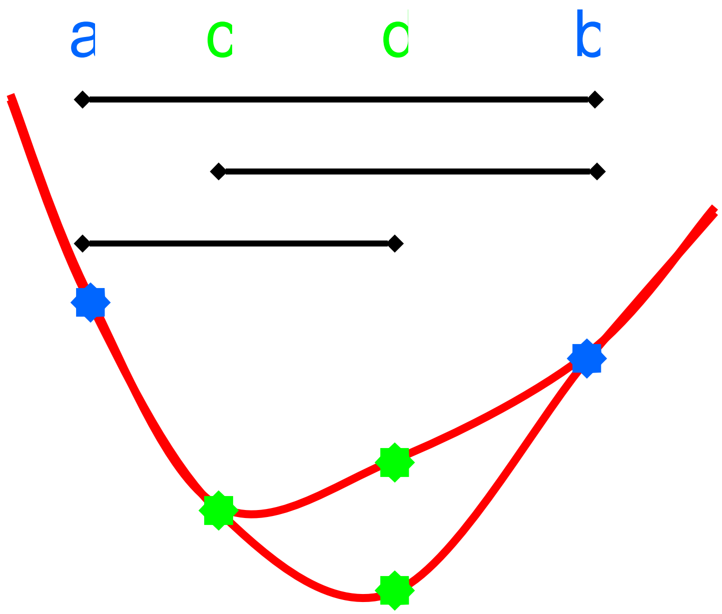
  [B] f(x1) > f(x2) > f(x3)

  [C] f is convex



- Then the minimum of f does not lie between x1 and x2
  - Can you prove this?

Creating a smaller interval



What should the spacing be?

- Either our 2nd interval is [a,d] which has c inside, or it's [c,b] and has d inside
  - In either case, we want to include this as part of the 3rd interval
    - To avoid doing an extra function evaluation
- If we want to keep the spacing of the points consistent between iterations, the relative distances involve the golden ratio
  - Algorithm is called *golden section search*

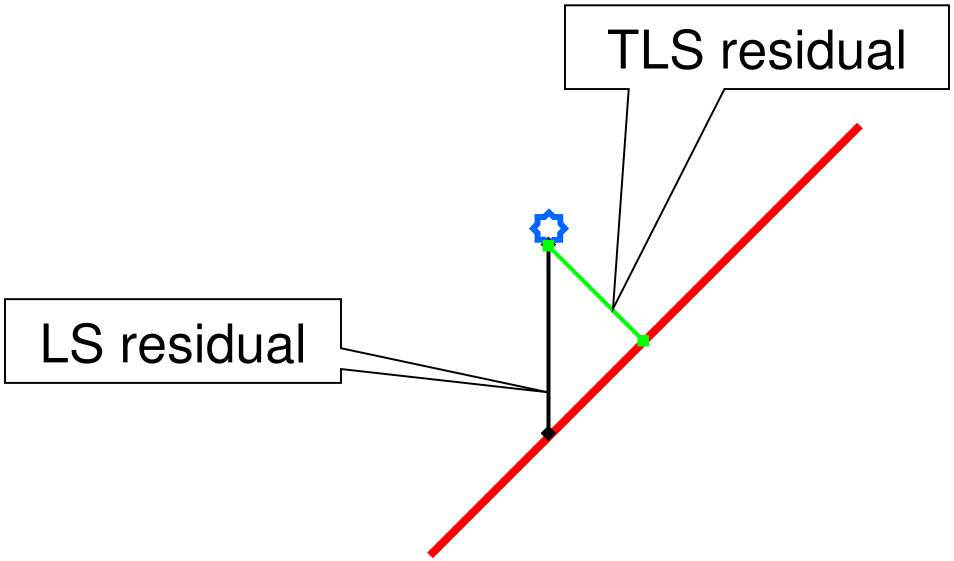
Another issue with LS fitting

- LS line fitting has an odd property, which is always a clue that something is wrong
  - It's not actually symmetric!
- If you interchange x and y, you get a different answer
  - Though it's usually close, in practice
  - This shows up in the terminology
    - Dependent versus independent variables
- The high school notion of least squares doesn't usually mention this fact

More about lying robots

- We assume that the robot tells the truth about the time, but lies about its position
  - What if the robot lies equally about both?
- This is actually quite realistic
  - Getting clocks synchronized precisely among different computers turns out to be hard
- Let's suppose that there are errors both in the time and in the position
  - Sometimes called "errors-in-variables"
  - Usually called: total least squares (Golub and van Loan, 1980)

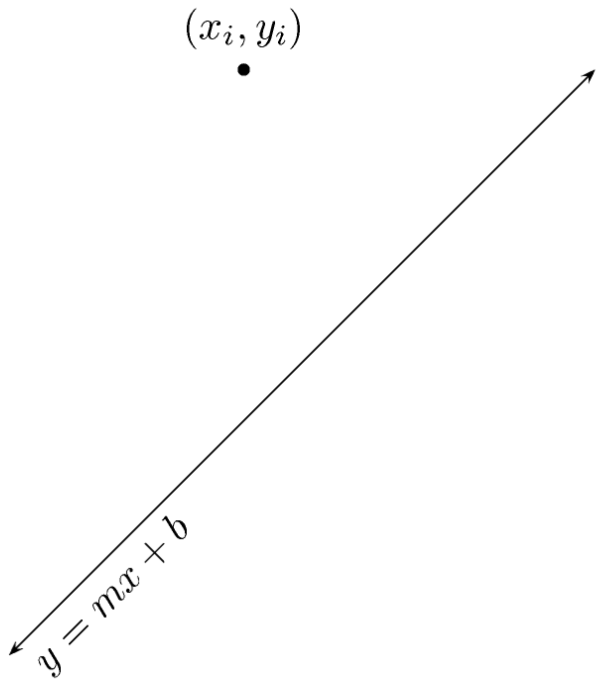
LS residual versus TLS residual



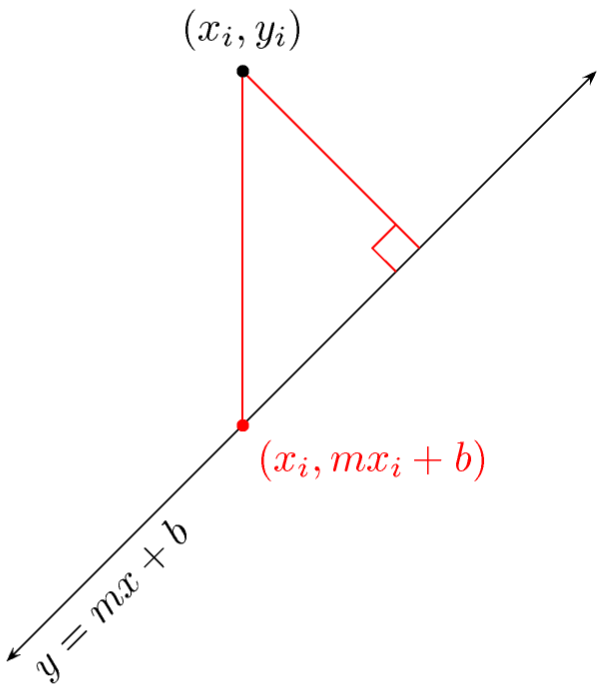
TLS line fitting

- We seek the line of best fit, where the error is the sum of the squared residuals
- The old residuals had a simple formula
  - "Vertical" distance from the point (x,y) to the line y=mx +b is y-(mx+b)
- Now we need to know: what is the distance from the point (x,y) to the line y=mx+b
  - Not the vertical distance! Distance to the closest point, instead.

Distance from point to line

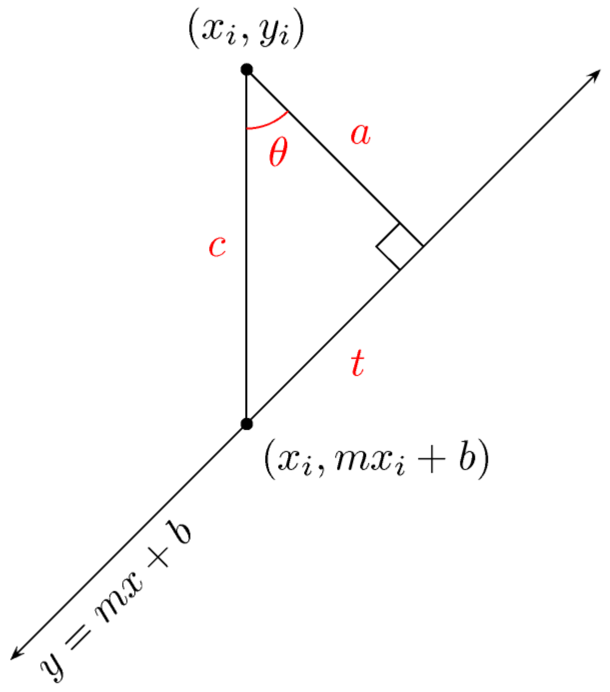


Shortest distance is perpendicular



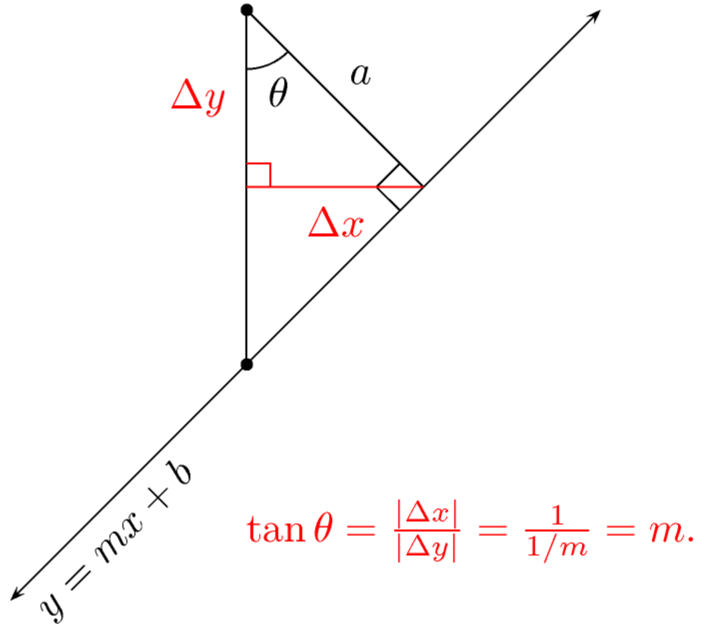
What do we know?

- We know the vertical distance
  - It's the regular LS residual



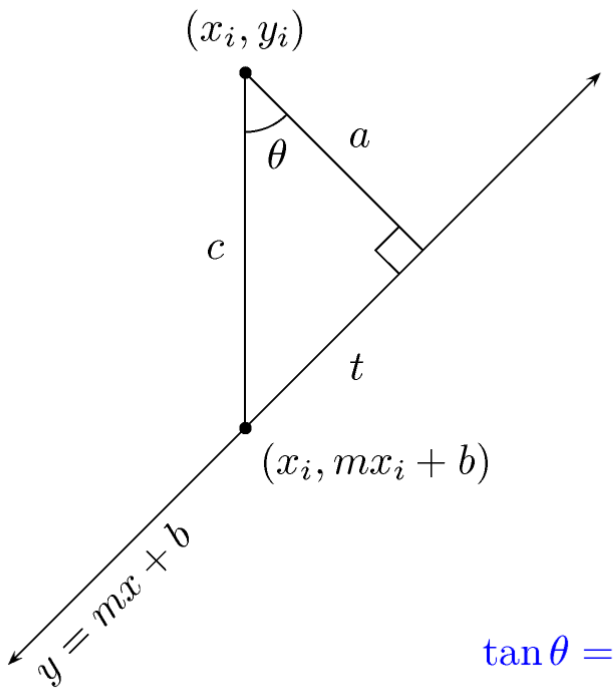
What else do we know?

- The slope of the line!



How does this help?

- We now know 2 sides of a right triangle
  - Time to invoke Pythagoras!



$$
\begin{aligned}
a &= \frac{t}{m} \\
&= \frac{\sqrt{c^2 - a^2}}{m} \\
&= \frac{c}{\sqrt{1 + m^2}} \\
&= \frac{y_i - mx_i - b}{\sqrt{1 + m^2}}.
\end{aligned}
$$

$$\tan\theta = m.$$