Conditioning and Stability

- Analogous concepts:
  - Conditioning of a *problem* = sensitivity to data errors
  - Stability of an *algorithm* = sensitivity to errors in computation

- Conditioning of a problem
  - problem solution is a map from input x to solution f(x)
  - PICTURE: error/uncertainty in data (x^), and error in solution (f(x^))
    - "backward error" $x - \hat{x}$
    - "forward error" $f(x) - f(\hat{x})$

- "well-conditioned" = insensitive
  "ill-conditioned" = sensitive

- How to make this notion *quantitative*?
  - ratio of relative forward error to relative backward error

$$K = \frac{\text{rel. forward err.}}{\text{rel. backward err.}} = \frac{|f(\hat{x}) - f(x)| \,/\, |f(x)|}{|\hat{x} - x| \,/\, |x|}$$

  - rearranging, see that K acts like "amplification factor"

$$\text{rel. forward err.} = K * \text{rel. backward err.}$$

  - ill-conditioned ---> large K
  - well-conditioned ---> small K or K close to 1

- Usually what we can derive is an upper bound for K, so that we get bound on rel. forward err.

$$\text{rel. forward err.} \le K * \text{rel. backward err.}$$

f is differentiable, $\hat{x} = x + dx$

$$f(x + dx) - f(x) \approx dx\, f'(x)$$

  - then K is

$$K_f = \frac{|dx\, f'(x)| \,/\, |f(x)|}{|dx| \,/\, |x|} = \frac{|f'(x)\, x|}{|f(x)|}$$

  - so K_f depends on properties of f and value of x

- There's a relationship between cond# of problem and cond# of inverse problem
  - Inverse problem of y = f(x) is find x s.t. f(x) = y, written x = f^-1(y) = g(y)
  - so

$$\frac{\text{rel. forward err.}}{\text{rel. backward err.}} = \frac{|g(\hat{y}) - g(y)| \,/\, |g(y)|}{|\hat{y} - y| \,/\, |y|} = \frac{|\hat{x} - x| \,/\, |x|}{|f(\hat{x}) - f(x)| \,/\, |f(x)|} = \frac{1}{K}$$

  - Differentiable f(x), and g(y)
    - $g(f(x)) = x$ by def'n
    - using chain rule, $g'(f(x))\ f'(x) = 1$, so g' = 1/f'
    - so cond#

$$K_g = \frac{|g'(y)\, y|}{|g(y)|} = \frac{|1/f'(x)\, f(x)|}{|x|} = \frac{1}{K_f}$$

  - Lesson:
    - If K_f near 1, both f and g well-conditioned
    - If K_f big or small, either K_f or K_g ill-conditioned

- Side note: Above is "relative cond#". If seeing x* s.t. f(x*) = 0, use "absolute cond#", defined analogously:

$$K = \frac{\text{abs. forward err.}}{\text{abs. backward err.}} = \frac{|f(\hat{x}) - f(x)|}{|\hat{x} - x|}$$

  - for differentiable f

$$K_{f\_abs} = \frac{|dx\, f'(x)|}{|dx|} = |f'(x)|$$

- Example: f(x) = sqrt(x) = x^{1/2}
  f'(x) = 1/2 * x^{-1/2} = 1/(2f(x))

$$K_f = \frac{|f'(x)\, x|}{|f(x)|} = \frac{|x|}{|2\, f(x) * f(x)|} = \frac{1}{2}$$

  - inverse problem: find x s.t. y = sqrt(x), or x = g(y) = y^2

    K_g = 2

  - Conclusion: both f and g are well-conditioned

- Example: f(x) = tan(x)
  f'(x) = sec^2(x) = 1 + tan^2(x)

$$K_f = \frac{|x(1+\tan^2(x))|}{|\tan(x)|} = \text{very large near } x = \pi/2$$

  - at x = 1.57079, K_f = 2.48275 * 10^5 (sensitive!!), so that

  tan(1.57079) ~= 1.58058 * 10^5, tan(1.57078) ~= 6.12490 * 10^4

((1.58058 * 10^5 - 6.12490 * 10^4) / (6.12490 * 10^4)) / ((1.57079 - 1.57078)/1.57078) = K_f

- $g(y) = \arctan(y)$, at $y = 1.58058 * 10^5$

  $K_g \sim= 4.0278 * 10^{-6}$ (insensitive!!)

Stability and Accuracy

- An algorithm is *stable* if its results are insensitive to perturbations during computation
  - e.g., truncation, discretization, and rounding errors

- Or, using backward error, algorithm is stable if
  - effect of perturbations during computation is no worse than effect of small amount of data error
  - *however* if problem is ill-conditioned, effect of small data error is really bad!
    - won't get a good (accurate) solution even with a stable algorithm

- So
  - well-conditioned problem + unstable algorithm => inaccurate solution
  - ill-conditioned problem + stable algorithm => inaccurate solution
  - well-conditioned problem + stable algorithm => accurate solution