Numerical Troubles

- Numerical difficulties can be encountered both in the solution of
  - Linear programs and
  - Nonlinear programs
- This lecture is about how to detect troubles and what to do when you have them

Why Numerical Troubles Occur

- Virtually every worthwhile numerical method related to optimization is based on the solution of linear systems ($Ax = b$) including
  - Linear programs
  - Nonlinear optimization problems
  - Complementarity problems
  - Least squares problems
  - Maximum likelihood problems

Error Analysis

- Essentially every real number that is stored on a computer is inaccurate due to allocation of a finite amount of storage to the number (Rounding)
- When calculations are performed, additional inaccuracies are introduced do to rounding of results
- One particularly bad type of inaccuracy that occurs is called cancellation in which two numbers that are nearly the same are differenced with the result having very few (if any) correct digits

Error Analysis (cont'd.)

- The vast majority of problems are not accurately calculated on a computer
- What measure of accuracy can we use?
- The most common measure is the 2-norm or Euclidean distance:

$$\|x\| = \sqrt{\sum^{n} x_i^2}$$

Error Analysis (cont'd.)

- A vector norm satisfies:
  - It is non-negative, equal to zero only if *x* is zero,
  - The norm of a scalar times a vector equals the absolute value of the scalar times the norm of the vector, $\|\alpha x\| = |\alpha| \|x\|$, and
  - The norm of the sum of two vectors is less than or equal to the sum of the norms of the vectors (The Triangle Inequality) $\|x + y\| \leq \|x\| + \|y\|$

Error Analysis (cont'd.)

- This idea can be extended to matrices as well – for instance, the Frobenius norm for a matrix is:

$$\|A\|_F = \sqrt{\sum^{m} \sum^{n} a_{ij}^{2}}$$

  - which has essentially the same properties as the vector 2-norm

- This pair of norms also satisfies "compatability"

$$\| \quad \| \quad \| \quad \| \quad \| \quad \|$$

Error Analysis (cont'd.)

- These concepts are useful for deriving what happens to the solution of a linear system when the data is somewhat inaccurate
- With exact arithmetic, the following is true:

$$Ax = b \quad \text{and} \quad x = A^{-1}b$$

Error Analysis (cont'd.)

- But what happens if the data for *b* is a little bit inaccurate?

$$^{-1}$$

- which means that $^{-1}$
- Applying norms to these expressions yields:

$$\|\delta x\| \le \|A^{-1}\|_F \|\delta b\| \quad \text{and} \quad \|b\| \le \|A\|_F \|x\|,$$

$$\text{or} \quad \frac{\|\delta x\|}{\|x\|} \le \|A\|_F \|A^{-1}\|_F \frac{\|\delta b\|}{\|b\|}$$

Error Analysis (cont'd.)

- What does this mean?

$$\frac{\|\delta x\|}{\|x\|} \leq \|A\|_F \, \|A^{-1}\|_F \, \frac{\|\delta b\|}{\|b\|}$$

- It means that the relative error in the solution to the linear system is equal to the relative error in the right-hand side times

$$cond(A) = \|A\|_F \|A^{-1}\|_F$$

- This number is often called the condition number of *A*

Error Analysis (cont'd.)

- So what?
- The "so what" is that when numerical methods fail, it is typically because either
  - The convergence conditions have been satisfied before a true optimum has been found, or
  - The condition number of the matrix in some linear system has become very large (and hence the linear system solutions are highly inaccurate)

Convergence Too Early

- "It is essential for a user to be suspicious about any pronouncements from an algorithm concerning the validity of a solution." (Gill, Murray, and Wright, <u>Practical Optimization</u>,1983)

Early Convergence (cont'd.)

- For an unconstrained (convex) problem
- The true solution satisfies ‖_ ‖
- Typically, no such *x* can be found (or would take too long to compute), and we must be satisfied with

‖_ ‖

- For a small value of epsilon

Early Convergence (cont'd.)

- Unfortunately, the level of epsilon embedded in algorithms is usually a fixed constant like 10^{-6}
- This is not a problem unless the norm of the gradient can be that small when *x* is a long way from the solution
- We had this problem with some of our expected utility maximization problems

Early Convergence (cont'd.)

- In general, consider the problem
- Ignoring the constraints, we would like the objective to be scaled so that its derivatives in the vicinity of the solution are not too small (or large)

Early Convergence (cont'd.)

- Assume we have a way of getting a ballpark guess of what *x* should look like at a solution
  - (E.g., for an expected utility problem we could set *x* to be equal to the expected profit maximizing solution)
  - We would like to choose a scaling factor for the objective function such that the derivatives of utility w.r.t. *x* at our approximate solution are near unity

Early Convergence (cont'd.)

- Let denote our approximate solution, and choose *a* so that

$$a\frac{\partial F(\hat{x})}{} \approx 1$$

- In the expected utility case,

$$\frac{\partial E[u(x,\beta)]}{} \approx \frac{\partial u(E(w))}{}\frac{\partial w}{} \approx \frac{\partial u(E(w))}{}$$

- (assuming that the final term in the second expression has good scaling)

Early Convergence (cont'd.)

- We then choose *a* so that

$$a\frac{\partial u(E(w))}{\partial w} \approx 1 \text{ or } a = \left[\frac{\partial u(E(w))}{\partial w}\right]^{-1}$$

- where for E[*w*], we substitute the expected return maximizing (risk neutral) level of wealth

When to Suspect Early Convergence

- It is difficult to recognize when convergence tests have been passed too early
- Here are some things that should make you feel more confident in a computed solution if true
  - Is the objective substantially smaller than it was at the starting point?
  - Do you converge to the same point if you change the starting point?
  - Do you converge to the same point if you use a different algorithm?

Multiple Starting Points

- What should you expect to find when you restart from multiple points in an effort to verify optimality?
  - If the problem is a convex program, then you should hope to match the optimal objective value, but not necessarily the optimal variable levels
  - If the problem is a *strictly* convex program, then you should hope to match the optimal objective value and variable levels

Scaling

- "The term 'scaling' is invariably used in a vague sense to discuss numerical difficulties whose existence is universally acknowledged, but cannot be described precisely in general terms." (Gill, Murray, and Wright, Practical Optimization, 1983)

Scaling (cont'd.)

- In its simplest form, scaling amounts to the selection of units for variables and equations
- Even with linear programs, choosing the scaling of variables and equations can make the difference between accurately solving a problem or not
- Rule of Thumb Try to choose units for variables and equations such that the largest entry in each row and column is with in a couple of orders of magnitude of unity

Scaling (cont'd.)

- For instance, in the ETA-Macro model of national energy planning for the U.S., nonelectric energy consumption is in units of quadrillions (10^{15}) of Btu's, and electric energy is in trillions (10^{12}) of kilowatt hours
- For problems with nonlinear constraints, the above rule of thumb applies to the Jacobian of the constraint functions

Scaling (cont'd.)

- It is also important to choose the units for the variables so that the objective is about "equally responsive" to changes in the variables – that is, the gradient is fairly well balanced
- This will also do a fair job of conditioning the Hessian matrix which is important for good algorithm performance
- (Some algorithms will print information regarding the conditioning of the Hessian – if you think you are having trouble, check into this)

Suspicions

- Finally, if your problems are taking a terribly long time to solve, you should become suspicious that something is wrong
  - Is the problem badly scaled?
  - Is the problem inefficiently formulated?
  - Is there a better choice of solver?

Good Luck, and Have Fun!

- Don’t be discouraged by all of the cautions and concerns that have been raised in this lecture
- Tools for formulation and solution of optimization problems have never been better or easier to use
- You now have the tools at your disposal to do work at the frontier of Optimization Modeling of Economic Systems

- May all your solutions be optimal!