Math 3607: Homework 11

(no due date)

You do not need to submit this assignment, yet you are highly encouraged to work out this problem set in preparation for Exam 4.

1 Optimal Step Size

In lecture, the optimal h for the second-order centered difference formula was shown to be about $\boxed{\text{eps}}^{1/3}$. At this optimal h, the leading error is $O(\boxed{\text{eps}}^{2/3})$. (Why?)

(a) (By hand) Determine the optimal h for the first-order forward difference formula by following a similar argument. Also determine the leading error at this optimal h.

(b) (By hand) Generalize the argument to determine the optimal h for an m-th order accurate method, where m is any positive integer. Also determine the leading error at this optimal h.

(c) (Computer) Complete the following program approximating the Jacobian of $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ using the first-order forward difference using the optimal step size determined in the previous parts.

```
function J = jacfd(f, x0)
% JACFD Approximation of a Jacobian by 1st-order forward difference
% Input:
%   f       function to be differentiated
%           which takes (n-by-1) column vector as an input
%           and produces (m-by-1) column vector as an output
%   x0      evaluation point
% Output:
%   J       approximate Jacobian (m-by-n)

    h = [.....];    % optimal step size





end
```

Hint. (**Tip for vectorization**) Recall that

$$\mathbf{J}(\mathbf{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \in \mathbb{R}^{m\times n} \tag{1}$$

The jth column of $\mathbf{J}$ consists of all partial derivatives with respect to x_j:

$$\mathbf{J}(\mathbf{x})\mathbf{e}_j = \begin{bmatrix} \frac{\partial f_1}{\partial x_j} \\ \frac{\partial f_2}{\partial x_j} \\ \vdots \\ \frac{\partial f_m}{\partial x_j} \end{bmatrix}$$

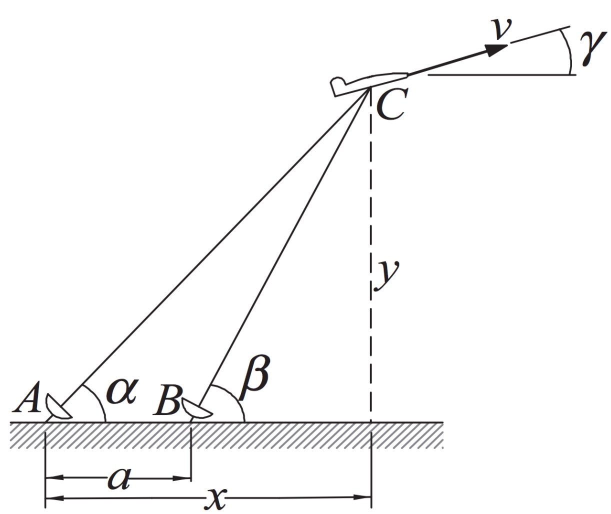
This column vector can be approximated by a finite difference formula involving a perturbation only in x_j:

$$\mathbf{J}(\mathbf{x})\mathbf{e}_j \approx \frac{\mathbf{f}(\mathbf{x} + h\mathbf{e}_j) - \mathbf{f}(\mathbf{x})}{h}, \quad j = 1, \ldots, n,$$

where h is optimally chosen according to the previous parts.

2 Air Plane Velocity from Radar Readings

(This exercise is adapted from an exercise in [1].) The radar stations A and B, separated by the distance $a = 500$ m, track a plane C by recording the angles α and β at one-second intervals. Your goal, back at air traffic control, is to determine the speed of the plane.



Let the position of the plane at time t be given by $(x(t), y(t))^{\mathrm{T}}$. The speed at time t is the magnitude of the velocity vector,

$$\left\| \frac{d}{dt} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} \right\| = \sqrt{x'(t)^2 + y'(t)^2}. \tag{2}$$

The closed forms of the functions $x(t)$ and $y(t)$ are unknown (and may not exist at all), but we can still use numerical methods to estimate $x'(t)$ and $y'(t)$. For example, at $t = 3$, the second order centered difference quotient for $x'(t)$ is

$$x'(3) \approx \frac{x(3+h) - x(3-h)}{2h} = \frac{1}{2}(x(4) - x(2)).$$

In this case $h = 1$ since data comes in from the radar stations at 1 second intervals.

Successive readings for α and β at integer times $t = 7, 8, \ldots, 14$ are stored in the file `plane.dat`. Each row in the array represents a different reading; the columns are the observation time t, the angle α (in degrees), and the angle β (also in degrees), in that order. The Cartesian coordinates of the plane can be calculated from the angles α and β as follows:

$$x(\alpha,\beta) = a\frac{\tan(\beta)}{\tan(\beta)-\tan(\alpha)} \quad \text{and} \quad y(\alpha,\beta) = a\frac{\tan(\beta)\tan(\alpha)}{\tan(\beta)-\tan(\alpha)}. \tag{3}$$

(a) (By hand) Verify the equations in (3).

(b) (Computer) Load the data, convert α and β to radians[1], then compute the coordinates $x(t)$ and $y(t)$ at each given t using (3). Approximate $x'(t)$ and $y'(t)$ using the second-order forward difference for $t = 7$, the second-order backward difference for $t = 14$, and the second-order centered difference for $t = 8, 9, \ldots, 13$. Return the values of the speed at each t using (2).

3 Visualization of Spectra and Pseudospectra

(This exercise is adapted from Chapter 7 of [2].) The eigenvalues of *Toeplitz* matrices, which have a constant value on each diagonal, have beautiful connections to complex analysis. Define six 64×64 Toeplitz matrices using

```
z = zeros(1,60);
A{1} = toeplitz( [0,0,0,0,z], [0,1,1,0,z] );
A{2} = toeplitz( [0,1,0,0,z], [0,2i,0,0,z] );
A{3} = toeplitz( [0,2i,0,0,z], [0,0,1,0.7,z] );
A{4} = toeplitz( [0,0,1,0,z], [0,1,0,0,z] );
A{5} = toeplitz( [0,1,2,3,z], [0,-1,-2,0,z] );
A{6} = toeplitz( [0,0,-4,-2i,z], [0,2i,-1,2,z] );
```

(The variable `A` constructed hereinabove is a *cell array*. See my HW08 solutions for an example involving cell arrays.) For each of the six matrices, do the following. This is a computer exercise entirely.

(a) Plot the eigenvalues of `A{#}` as red dots in the complex plane. (Set `'MarkerSize'` to be 3.)

(b) Let E and F be 64×64 random matrices generated by `randn`. On top of the plot from part (a), plot the eigenvalues of the matrix $A + \varepsilon E + i\varepsilon F$ as blue dots, where $\varepsilon = 10^{-3}$. (Set `'MarkerSize'` to be 1.)

(c) Repeat part (b) 49 more times (generating a single plot).

Arrange all six plots in a 3×2 grid using `subplot`. Make sure all figures are drawn in 1:1 aspect ratio.

4 Vandermonde Matrix, SVD, and Rank

Let $\mathbf{x}$ be a vector of 1000 equally spaced points between 0 and 1, and let A_n be the $1000 \times n$ Vandermonde-type matrix whose (i, j) entry is x_i^{j-1} for $j = 1, \ldots, n$. This is a computer exercise.

[1] You may ignore this step and use `tand`.

(a) Print out the singular values of A_1, A_2, and A_3.

(b) Make a semi-log plot of the singular values of A_{25}.

(c) Use `rank` to find the rank of A_{25}. How does this relate to the graph from part (b)? You may want to use the help document for the `rank` command to understand what it does.

5 SVD and 2-Norm

Let $A \in \mathbb{C}^{m \times n}$ have an SVD $A = USV^*$. The following problem walks you through the proof of the fact that $\|A\|_2 = \sigma_1$. Do this by hand.

(a) Use the technique of Lagrange multipliers to show that among vectors that satisfy $\|\mathbf{x}\|_2^2 = 1$, any vector that maximizes $\|A\mathbf{x}\|_2^2$ must be an eigenvector of A^*A.

(*Hint.* If B is any hermitian matrix, *i.e.*, $B^* = B$, the gradient of the scalar function $\mathbf{x}^*B\mathbf{x}$ with respect to $\mathbf{x}$ is $2B\mathbf{x}$.)

(b) Use the result of part (a) to prove that $\|A\|_2 = \sigma_1$, the *principal singular value* of A.

References

[1] Jaan Kiusalaas. *Numerical methods in engineering with Python 3*. Cambridge university press, 2013.

[2] Lloyd N. Trefethen and Mark Embree. *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*. Princeton University Press, 2005.