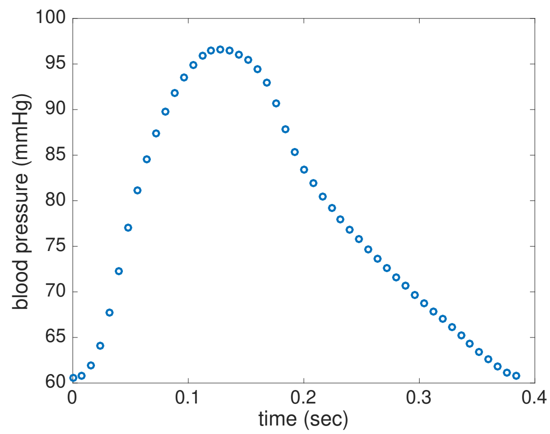
Math 3607: Exam 2 (Written)

Due: 11:59PM, Friday, March 5, 2021

1 Least Squares for Periodic Data [25 points]

The graph below represents arterial blood pressure collected at 8 ms intervals (over one heart beat) from an infant patient:



Denote the data points by (t_i, y_i) for $i = 1, \dots, m$. The data can be fit using a low-degree polynomial of the form

$$f(t) = c_1 + c_2 t + \cdots + c_n t^{n-1}, \quad n < m. \tag{1}$$

In the most general terms, the fitting function takes the form

$$f(t) = c_1 f_1(t) + \cdots + c_n f_n(t), \tag{2}$$

where $f_1, \dots, f_n$ are known functions while $c_1, \dots, c_n$ are to be determined to optimize the fit to the data. This optimization can be formulated as an $m \times n$ LLS problem of minimizing the 2-norm of the residual $\|\mathbf{y} - A\mathbf{c}\|_2$, where $A_{i,j} = f_j(t_i)$.

(a) Download the data file `pressuredata.mat` and load them into MATLAB using

```
load pressuredata
```

This creates two vectors `t` and `y` containing time and blood pressure data, respectively. Use them to regenerate the plot above.

(b) Fit the data to a straight line, $f(t) = c_1 + c_2 t$. Solve for the coefficients using backslash. Superimpose the graph of the fitting line on your graph from the previous step, and compute the 2-norm of the residual $\|\mathbf{y} - A\mathbf{c}\|_2$, where $A_{i,j} = t_i^{j-1}$ is a *Vandermonde*-type matrix.

(c) Repeat part (b) for a quadratic and a cubic polynomial. The residual norm will get smaller in each case, but there is still a room for improvement.

(d) Exploiting the fact that the data come from a periodic phenomenon (heart beats), adapt (2) to a periodic fitting function

$$f(t) = c_1 + c_2 \cos \frac{2\pi t}{T} + c_3 \sin \frac{2\pi t}{T} + c_4 \cos \frac{4\pi t}{T} + c_5 \sin \frac{4\pi t}{T}, \quad \text{where } T = t_m - t_1. \tag{3}$$

As in the previous parts, solve for the coefficients using backslash, superimpose the graph of $f(t)$ to the plot of data points, and compute the residual norm. Comment on your observation.

Template.

```
%% Part (a): Load and plot data -------------------------------------------
load pressuredata
% Make a labeled plot.




%% Part (b): Fit to a straight line ---------------------------------------
% Set up matrix A and solve for c using backslash.




% Compute the residual norm.




% Plot




%% Part (c): Fit to quadratic and cubic fits ------------------------------







%% Part (d): Fit to a periodic function -----------------------------------
% Create and plot a fit from periodic functions. Compute the residual norm.







%-------------------------------------------------------------------------
```

2 Visualization of Matrix Norms (3-D) [25 points]

Recall that

$$\|A\|_p = \max_{\|\mathbf{x}\|_p=1} \|A\mathbf{x}\|_p, \quad p \in [1, \infty].$$

In this problem, we generate three-dimensional visualization of this definition.

(a) Complete the following program which, given $p \in [1, \infty]$ and $A \in \mathbb{R}^{3\times3}$, approximates $\|A\|_p$ and plots the unit sphere in the p-norm and its image under A.

```
function norm_A = visMatrixNorms3D(A, p)
    %% Basic checks
    if size(A,1)~=3 || size(A,2)~=3
        error('A must be a 3-by-3 matrix.')
    elseif p < 1
        error('p must be >= 1.')
    end

    %% Step 1: Initialization
    nr_th = 41; nr_ph = 31;
    th = linspace(0, 2*pi, nr_th);
    ph = linspace(0, pi, nr_ph);
    [T, P] = meshgrid(th, ph);
    x1 = cos(T).*sin(P);
    x2 = sin(T).*sin(P);
    x3 = cos(P);
    X = [x1(:), x2(:), x3(:)]';

    %% Step 2: [FILL IN] Normalize columns of X into unit vectors




    %% Step 3: [FILL IN] Form Y = A*X and then calculate norms of columns of Y




    %% Step 4: [FILL IN] Calculate p-norm of A (approximate)




    %% Step 5: [FILL IN] Generate surface plots




end
```

(The function must be written at the very end of your Live Script.)

The following steps are carried out by the program.

- **Step 1:** Create 3-vectors

$$\mathbf{x}_k = \begin{bmatrix} \cos\theta_i \sin\phi_j \\ \sin\theta_i \sin\phi_j \\ \cos\phi_j \end{bmatrix}, \quad \text{for } 1 \leqslant i \leqslant 41,\ 1 \leqslant j \leqslant 31 \tag{4}$$

  using 41 evenly distributed θ_i in $[0, 2\pi]$ and 31 evenly distributed ϕ_j in $[0, \pi]$. Note the use of `meshgrid`, which is useful for surface plots later.

- **Step 2:** Normalize $\mathbf{x}_k$ into a unit vector in p-norm by $\mathbf{x}_k \to \mathbf{x}_k / \|\mathbf{x}_k\|_p$.
- **Step 3:** For each k, let $\mathbf{y}_k = A\mathbf{x}_k$. Calculate and store $\|\mathbf{y}_k\|_p$.
- **Step 4:** Approximate $\|A\|_p$ based on the norms $\|\mathbf{y}_k\|_p$ calculated in the previous step.
- **Step 5:** Generate surface plots of the unit sphere in the p-norm and its image under A. Use `surf` function; see Lecture 8. Use `subplot` to put two graphs side by side.

(b) Run the program with $p = 1, \frac{3}{2}, 2, 4$, all with the same matrix

$$A = \begin{bmatrix} 2 & 0 & 0 \\ 0 & \cos(\pi/12) & -\sin(\pi/12) \\ 0 & \sin(\pi/12) & \cos(\pi/12) \end{bmatrix}. \tag{5}$$

```
A = [2 0 0;
     0 cos(pi/12) -sin(pi/12);
     0 sin(pi/12)  cos(pi/12)];
visMatrixNorms3D(A, 1);
visMatrixNorms3D(A, 3/2);
visMatrixNorms3D(A, 2);
visMatrixNorms3D(A, 4);
```

(Depending on how you write the code, you may need to use `clf` or `hold off` in between function calls.)

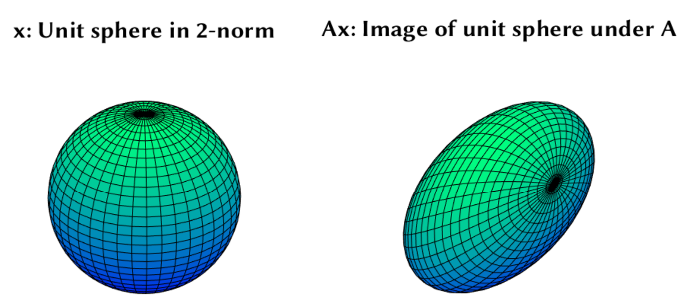


Figure 1: Example output.

3 Extra Credit [5 points]

Do **LM** 7.2–13. (This was an optional problem assigned for Homework 5.)
The figures below are all generated with `level=6`.

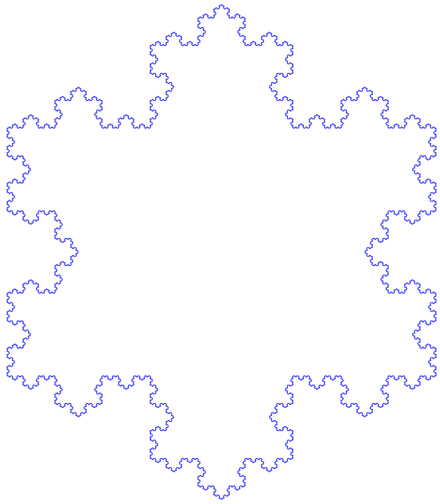


Figure 2: Koch curves around a *negatively* oriented triangle. Generated by letting the angles be $0, -120$, and -240.

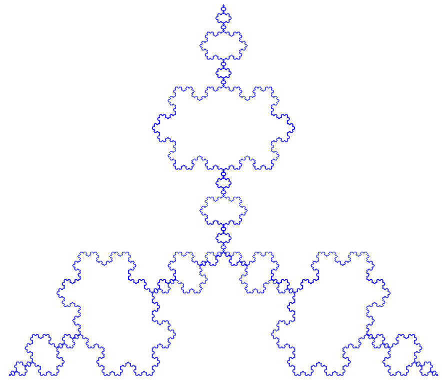


Figure 3: Koch curves around a *positively* oriented trianlge. Generated by letting the angles be $0, 120$, and 240.

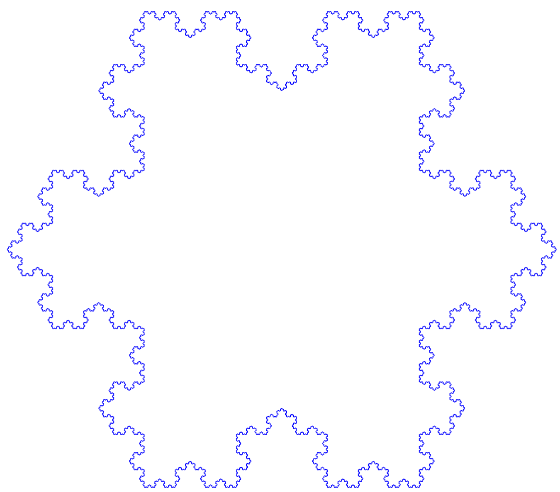


Figure 4: Koch curves around a hexagon. Generated by letting the angles be $0, 60, \ldots, 300$.