Math 3607: Homework 6

Due: 11:59PM, Tuesday, March 2, 2021

TOTAL: 20 points

1. Do **LM** 12.5–3. (Understanding matrix multiplication)

2. Do **LM** 12.6–2. (Gram-Schmidt)

3. (Adapted from **FNC** 3.3.3.) Let $x_1, x_2, \ldots, x_m$ be m equally spaced points in $[-1, 1]$. Let V be the Vandermonde-type matrix appearing on p.10 of Lecture 17 slides for $m = 400$ and $n = 5$. Find the thin QR factorization of $V = \widehat{Q}\widehat{R}$, and, on a single graph, plot every column of $\widehat{Q}$ as a function of the vector $\mathbf{x} = (x_1, x_2, \ldots, x_m)^{\mathrm{T}}$. (Use MATLAB to solve this problem.)

4. (Visualizing matrix norms; adapted from **LM** 9.4–26.) For $p \in [1, \infty]$, recall the definition of the matrix p-norm,
$$\|A\|_p = \max_{\|\mathbf{x}\|_p = 1} \|A\mathbf{x}\|_p .$$
To understand this definition, we will work in two-dimensional space so that we can easily plot the results. For this problem, use
$$A = \begin{bmatrix} 2 & 1 \\ 1 & 3 \end{bmatrix}. \tag{1}$$
As an illustration, we study the case $p = 2$ following the steps below.

   - Create unit vectors $\mathbf{x}_j$ in 2-norm,
   $$\mathbf{x}_j = \begin{bmatrix} \cos\theta_j \\ \sin\theta_j \end{bmatrix}, \quad 1 \leqslant j \leqslant 361 \tag{2}$$
   using 361 evenly distributed θ_j in $[0, 2\pi]$. Make sure $\mathbf{x}_1 = \mathbf{x}_{361} = (1, 0)^{\mathrm{T}}$, just as in the spiral polygon problem. Plot these points, which lie on the unit circle. Make sure the plot looks like a circle.
   - For each j, let $\mathbf{y}_j = A\mathbf{x}_j$. Plot all points $\mathbf{y}_j$. In addition, store $\|\mathbf{y}_j\|_2$ for all j in a vector.
   - Plot $\|\mathbf{y}_j\|_2$ as a function of θ_j.
   - Find the maximum value of $\|\mathbf{y}_j\|_2$ over all j. This estimates $\|A\|_2$. Compare this against the actual value computed by `norm(A,2)`.

   These steps are carried out by the following script.

```
A = [2 1; 1 3];
theta = linspace(0, 2*pi, 361);
X = [cos(theta); sin(theta)];
Y = A*X;
norm_Y = sqrt(sum(Y.^2, 1));
```

```
clf
subplot(2,2,1)
plot(X(1,:), X(2,:)), axis equal
title('x: unit vectors in 2-norm')
subplot(2,2,2)
plot(Y(1,:), Y(2,:)), axis equal
title('Ax: image of unit vectors under A')
subplot(2,1,2)
plot(theta, norm_Y), axis tight
xlabel('\theta')
ylabel('||y||')
fprintf(' p = 2\n')
fprintf(' approx. norm: %18.16f\n', max(norm_Y))
fprintf(' actual  norm: %18.16f\n', norm(A, 2))
```

which generates Figure 1 and the following outputs in the Command Window:

```
p = 2
approx. norm: 3.6179964204609893
actual  norm: 3.6180339887498953
```

Modify the script to carry out the same tasks for $p = 1, \infty$. Pay particular attention to the lines where `X` is defined and `norm_Y` is calculated.

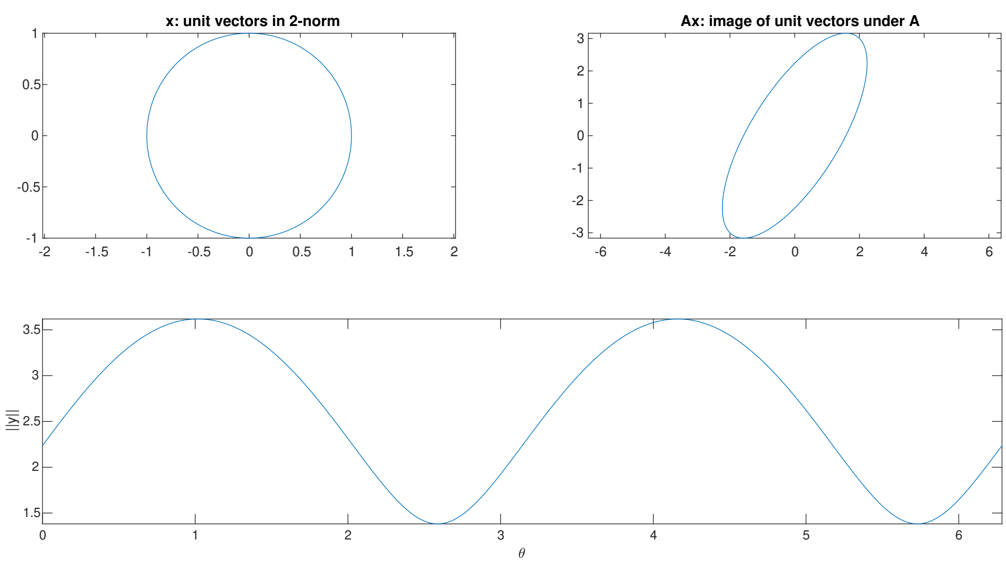

Figure 1: Plots illustrating the definition of matrix norm.

5. **Graphics exercise of the week: Julia and Mandelbrot Sets**

   This is an *optional* problem for those interested in further developing programming skills and creating cool graphics. Read **LM** 6.8.3 on Julia and Mandelbrot sets. Then generate fractals by working out **LM** 6.8–69.