MTH 451/551 – Lab 1

1. (30 points) Write an algorithm for matrix-vector multiplication $\vec{b} = A\vec{x}$ in MATLAB using two different ways:

   (a) by computing inner products of rows and columns

   (b) by representing the product as a linear combination of columns of A.

   For each case, test your MATLAB code for some random matrices of size $m \times m$ and some random m-vector. In particular, test the following:

   (i) time each method by adding `tic` before the `for` loop and `toc` after

   (ii) test that the results are essentially equivalent (to round-off) by outputing some measure of the *relative* distance between the products

   (iii) use $m = 2$ and $m = 100$ in your comparisions.

   Turn in your source code and the output it produces. Explain in words what you observe. Note: whenever using random matrices, be sure to run each case multiple times to avoid the possibility of a fluke!

2. (30 points) Write an algorithm for finding the residual of the "best" approximation to a vector $\vec{x}$ in the space spanned by n orthonormal m-vectors $\{\vec{q}_i\}$ in MATLAB using two different ways, (as defined by solving Equation 2.7 in Trefethan-Bau for $\vec{r}$)

   (a) $\vec{r} = \vec{v} - \sum_{i=1}^{n} (\vec{q}_i^* \vec{v})\, \vec{q}_i$

   (b) $\vec{r} = \vec{v} - \sum_{i=1}^{n} (\vec{q}_i \vec{q}_i^*)\, \vec{v}$.

   For each case, test your MATLAB code for some random orthonormal vectors (try `Q=orth(rand(m,n))`) and some random m-vector. In particular, test the following:

   (i) time each method by adding `tic` before the `for` loop and `toc` after

   (ii) test that the results are essentially equivalent (to round-off) by outputing some measure of the *relative* distance between the results

   (iii) use $(m, n) = (50, 30)$ and $(m, n) = (50, 50)$ in your comparisions.

   Turn in your source code and the output it produces. Explain in words what you observe. Note: be sure to run each case multiple times.

3. Write a function `ball` which plots the unit ball corresponding to the p-norm for a given p (or download **ball.m** from the course website). Verify the plots in Equation 3.2 of Trefethan-Bau. Do not turn in.

4. (40 points) Modify the m-file from the previous exercise to create a function `Aball` which takes as an additional input a matrix A and plots the image of the unit ball under the mapping defined by A.

   (a) Verify the plots in Figure 3.1 corresponding to the matrix

   $$\begin{bmatrix} 1 & 2 \\ 0 & 2 \end{bmatrix}.$$

   Do not turn in.

   (b) One can define a notion of the *condition* of a matrix based on how "skinny" its image of the unit ball is (say in the 2-norm).

      i. Run `A=rand(2);Aball(A)` a few times.

      ii. When you see a particularly skinny one, display the following:

         `A`

         `det(A)`

         `eig(A)`

         `cond(A)` (we will define this later)

      iii. Do the same for a particularly fat one.

      iv. Discuss the possible relation between the condition of a matrix, the eigenvalues, and the determinant.

   Turn in your source code and selected output, including plots.