Assignment #7

Due Wednesday, 28 October, 2015 at the start of class

Please read Lectures 8, 9, 10, 11 in Trefethen & Bau.

P16. *Note that Exercise 8.2 below asks you to implement Algorithm 8.1 in* MATLAB*. Exercise 10.2 asks you to implement Algorithm 10.1. Do those Exercises first.*

(a) Implement a MATLAB function `[Q,R] = clgs(A)` which computes the reduced QR decomposition of $A \in \mathbb{C}^{m\times n}$, in the case $m \geq n$, using the classical Gram-Schmidt orthogonalization, Algorithm 7.1.

(c) Test

- `clgs()`
- `mgs()` from Exercise 8.2
- `house()` & `formQ()` from Exercise 10.2
- `qr()` the MATLAB built-in[1]

on a random matrix `A = randn(50,12)`. Specifically, use

- `norm(Q'*Q - eye(n))`
- `norm(tril(R,-1))`
- `norm(Q*R - A)/norm(A)`

to verify unitary-ness, upper-triangularity, and factorization, respectively, in each case. Thus for each method in the first list, generate the three numbers from the second list. (*Make a nice table of results from this experiment.*) The numbers would all be zero if everything were exact. Do this experiment twice to show that the random matrix was not special.

(d) The authors claim Algorithms 8.1 and 10.1 have superior stability properties to Algorithm 7.1. Do you see evidence for this claim from the part (c) results? Compare to the results from Experiments 2 and 3 in Lecture 9.

Exercise 8.1 in Lecture 8.

Exercise 8.2 in Lecture 8.

Exercise 9.2 in Lecture 9. (*Additional hint on* (c): Use a result from Exercise 3.3.)

Exercise 10.1 in Lecture 10.

Exercise 10.2 in Lecture 10.

Exercise 11.3 in Lecture 11.

[1] Ask `qr()` for the *reduced* factorization, of course.