MATH 4310/8510
Fall 2014
Assignment 2

Due Friday, October 31^{st}, 2014

General instructions. The assignment is due Friday, October 31^{st}, 2014 before closing of the office of the Department of Mathematics (16:30) or in pdf form before midnight (CST). Late assignments will **not** be accepted. This assignment is common to the undergraduate and graduate versions of the course.

---

1 Mathematical questions

This part is worth 70 marks.

- This is a fourth-year/grad math class. It is expected that your answers will be clear yet succinct.
- I encourage you to work with others on these problems. However, the solution that you hand in **must** be yours; copying is not permitted and will be considered to be relevant of Academic Dishonesty. Papers with large numbers of corresponding problems treated will be scrutinized, so beware!
- Attempt all questions.

1. Consider the matrix

$$A = \begin{pmatrix} 2 & -\frac{1}{2} & \frac{1}{4} & -\frac{1}{4} \\ \frac{1}{4} & 1+2i & 0 & \frac{1}{4} \\ -\frac{1}{2} & \frac{1}{4} & -1 & \frac{1}{2} \\ \frac{1}{4} & -\frac{1}{2} & \frac{1}{2} & -2-2i \end{pmatrix}$$

State and draw the Gershgorin disks associated to A. Compute the eigenvalues of A and indicate them in your plot. Discuss the use of the theorem about the localization of the eigenvalues in the different sets of Gershgorin disks.

2. A square matrix is diagonalizable if it is similar to a diagonal matrix.. Find and justify an "algorithm" that allows you to decide on the diagonalizability of a matrix and provides, when it is diagonalizable, the similarity transformation and the diagonal matrix. (It is expected that you will seek such an algorithm from books/online references, not that you will develop it. Cite your source(s).)
Use your algorithm to consider the diagonalizability of the following matrices.

$$B_1 = \begin{pmatrix} 0 & 1 & 0 \\ 4 & 4 & 0 \\ 2 & 1 & 2 \end{pmatrix}, B_2 = \begin{pmatrix} 3 & 0 & 1 \\ 2 & 4 & 2 \\ 1 & 0 & 3 \end{pmatrix}.$$

3. For which values of a is the matrix

$$C_a = \begin{pmatrix} 0 & 1 \\ -a & 1+a \end{pmatrix}$$

diagonalizable? When it is, find the similarity transformation and the diagonal matrix associated to C_a.

4. In the cases where the matrices are not diagonalizable in Exercises **2** and **3**, give the Jordan normal form of the matrices.

5. Which of the following matrices is irreducible? Primitive? (State the definitions and theorems you are using.) If a matrix is reducible, give a reduced (block triangular) form of the matrix.

$$D_1 = \begin{pmatrix} 1 & 0 & 2 & 0 & 3 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 4 & 0 & 0 & 0 \end{pmatrix}, D_2 = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, D_3 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}.$$

2 "Computer algebra" questions

We continue with the MatLab/Octave "matrix toolkit". This part is worth 30 marks.

Instructions:

- Use MatLab or Octave.
- The solution will consist of two parts: a companion pdf file with, where needed, documentation of the functions produced, stating explicitly the definitions and theorems used, etc.; a set of MatLab/Octave files, each corresponding to a function. You can forgo the companion pdf if your code is well commented.
- The functions/files shall be named as prescribed here, so that a single test script can be used to verify the results for all students. A sample test script is available on the course website and should be used to ensure in particular that errors in the script do not interrupt the execution of the script.
- Create a folder named in the format Name_StudentID_A02, put all `.m` files there as well as the pdf of the documentation (and if you are turning in the rest of the assignment electronically -for instance scanned or typed-, the corresponding pdf file) and 7zip, zip or rar the folder.

First, we refine the functions already produced in assignment 1, making the return values more consistent:

- 1 should be returned when the property is true.
- 0 should be returned when the property is false.
- -1 indicates that the property is true with the generalized definition of a property. For instance, triangular and diagonal matrices need not be square.
- -2 indicates a dimension problem not covered by the -1 case. The empty matrix [ ] should also return -2. So should any property defined exclusively on square matrices, etc. For example,

  $$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

  is not diagonal but is "extended diagonal", so `is_diagonal_matrix` should return -1 (and so should the functions testing for triangularity). On the other hand, the matrix

  $$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

  fails to satisfy both the "classic" and "extended" definitions of diagonality/triangularity and should return -2.

Given a matrix $M \in \mathcal{M}_{mn}$, the following functions should be available:

- `is_real_matrix`(M) is true if $M \in \mathcal{M}(\mathbb{R})$.
- `has_imaginary_entries`(M) is true if $M \in \mathcal{M}(\mathbb{C})$ has some imaginary entries.
- `is_diagonal_matrix`(M) is true if $M \in \mathcal{M}(\mathbb{C})$ is a diagonal matrix.
- `is_lower_triangular_matrix`(M) is true if $M \in \mathcal{M}(\mathbb{C})$ is a lower triangular matrix.
- `is_upper_triangular_matrix`(M) is true if $M \in \mathcal{M}(\mathbb{C})$ is an upper triangular matrix.
- `is_triangular_matrix`(M) is true if $M \in \mathcal{M}(\mathbb{C})$ is a triangular matrix.
- `is_hermitian_matrix`(M) is true if $\mathcal{M} \in \mathcal{M}(\mathbb{C})$ is a Hermitian matrix.

Additionally, for $M \in \mathcal{M}_n(\mathbb{R})$, create the following functions, using the additional error code -3 in the case where the matrix is square but has imaginary entries.

- `is_nonnegative_matrix`(M) is true if $M \geq 0$.
- `is_positive_matrix`(M) is true if $M \gg 0$.
- `is_irreducible_matrix`(M) is true if M is irreducible.
- `is_primitive_matrix`(M) is true if M is primitive.
- `primitivity_index`(M) returns the usual error codes, 0 if the matrix is square and real but not primitive and the primitivity index if M is primitive, i.e., the smallest k such that $A^k \gg 0$.

For the latter three, "shop around" in the textbook (and elsewhere), there are some results that could be helpful.

Finally, create a function `matrix_properties`(M) returning a structure with fields having the same names as all the functions you created as storing the return value of these functions. For instance, if M is primitive and that your call to function is `p=matrix_properties(M)`, then `p.is_primitive_matrix=1`.