MATH 210 Assignment 6

Linear algebra

INSTRUCTIONS

- Create a new Python 3 Jupyter notebook
- Answer each question in the Jupyter notebook and clearly label the solutions with headings
- Functions should include documentation strings and comments
- There are 20 total points and each question is worth 4 points
- Submit the `.ipynb` file to Connect by **11pm Tuesday, March 21, 2017**
- You may work on these problems with others but you must write your solutions on your own

QUESTIONS

1. Write a function called `a_solve` which takes an input parameter a and returns the solution (as a 1D NumPy array) of the system of equations

$$\begin{bmatrix} 4 & 5 & 7 \\ 0 & 1 & 9 \\ 1 & 3 & 7 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a \\ 1-a \\ 1 \end{bmatrix}$$

2. Write a function called `minor_det` which takes input parameters A, i and j and returns the determinant of the matrix $A_{i,j}$ obtained from the square matrix A by removing the ith row and jth column of A. The indices i and j start at 0. For example, if the matrix A is

$$A = \begin{bmatrix} 4 & 5 & 7 \\ 0 & 1 & 9 \\ 1 & 3 & 7 \end{bmatrix}$$

   then the matrix $A_{0,1}$ is

$$A_{0,1} = \begin{bmatrix} 0 & 9 \\ 1 & 7 \end{bmatrix}$$

3. Write a function called `outer_eig` which takes input parameters $\mathbf{u}$ and $\mathbf{v}$ (1D NumPy arrays of the same length) and returns the eigenvalues (as a 1D NumPy array) of the outer product $\mathbf{u}^T\mathbf{v}$ (where $\mathbf{u}^T$ is a column vector and $\mathbf{v}$ is a row vector).

4. Write a function called `elementary` which takes input parameters A, λ, i, j and `row` (default value `row=True`) and performs the following tasks:

(a) If `row=True`, return the matrix obtained from A by adding λ times the ith row to the jth row

(b) If `row=False`, return the matrix obtained from A by adding λ times the ith column to the jth column

5. Write a function called `nearest_point` which takes input parameters `coeffs` and `P` where `coeffs` $= [a, b, c, d]$ is a list of numbers representing a plane

$$ax + by + cz = d$$

and `P` $= [x, y, z]$ is a list of length 3 representing a point in $\mathbb{R}^3$, and the function returns the point Q (as a 1D NumPy array of length 3) on the plane which is nearest to the point `P`.