[MATH2605] Homework 10

Due: Apr. 6

Jacobi Iteration

Consider a symmetric matrix

$$\mathbf{A} = \begin{bmatrix} 4 & -1 & -3 & -1 \\ -1 & -2 & -1 & -5 \\ -3 & -1 & -4 & -3 \\ -1 & -5 & -3 & 2 \end{bmatrix}. \tag{1}$$

Problem 1

Using the Matlab code provide below, report **B** matrix after the first five iterations.

Problem 2

Now, we want to add a stopping condition

$$\mathrm{Off}(B) \equiv \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} |B_{ij}^2| \quad \leq \quad \frac{1}{n-1}\left(\frac{\varepsilon}{2n-1}\right)^2, \tag{2}$$

which is the equation (3.19). Note that since **A** is a 4×4 matrix, $n = 4$. Check the condition (2) at the end of each iteration, and stop the iteration if (2) is satisfied. Note that this is to stop the iteration when each entry of **B** is within ε distance from some eigenvalue of **A**. Also note that **A** and **B** have same eigenvalues.

- How many iterations do you need when $\varepsilon = 0.01$?
- How many iterations do you need when $\varepsilon = 0.001$?
- How many iterations do you need when $\varepsilon = 0.0001$?

Problem 3

Finally, we would like to compare the diagonals of **B** and the eigenvalues of **A**. Compute all eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ by the *eig(A)* command. We consider these λ_j, $j = 1, 2, \ldots, n$ as exact eigenvalues. Define the error E by

$$E = \max_{i=1,2,\ldots,n} \; \min_{j=1,2,\ldots,n} |\mathbf{B}_{ii} - \lambda_j|, \tag{3}$$

Note that E is the maximum of differences between the diagonal elements $\mathbf{B}_{ii}$'s and eigenvalues of **A** that are closest to the $\mathbf{B}_{ii}$'s. Report E for the first 15 iterations. Does the condition (2) indeed stops the iteration when $E < \varepsilon$.

I am providing a Matlab code that implements the Jacobi iteration.

```
A = [4    -1    -3    -1
    -1    -2    -1    -5
    -3    -1    -4    -3
    -1    -5    -3     2];
n = size(A,1);
eig(A)

B = A;
V = eye(n);

for ix=1:1000
    % Find maximum off-diagonal entry
    Bu = triu(abs(B)) - diag(diag(abs(B)));
    [P,IX1] = max(Bu);
    [P,IX2] = max(P);
    ij = [IX1(IX2), IX2];
    % compute eigenvalues and eigenvectors of the 2x2 sub-matrix.
    [U,D] = eig(B(ij,ij));
    % construct n-by-n rotation matrix G
    G = eye(n); G(ij,ij) = U;
    % Update B and G
    B = G'*B*G
    V = V*G;
end
```