Numerical Analysis — Exam Solutions — FMNF05 2019/03/21

The exam lasts 4 hours and has 3 sections in 4 pages.
A minimum of 26 points out of the total 53 are required to get a passing grade of 3 (assuming you have already passed the assignments).
To get a 4 you need at least 35 points. Finally to get a 5 you need at least 44 points.

During the exam you are allowed one page (both sides) with hand-writen notes. No calculators, textbooks, lecture notes or any other electronic or written material is allowed.

Section A. (9 pts) True/False Section.
No justification needed. Just write/circle either True or False after each question.

A.1 The number of operations to solve $Ax = b$ using the inverse A^{-1} is approximately equivalent to using the LU factorization. (**T** / F)

A.2 The matrix $A = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 2 & 0 \\ 3 & 1 & 0 \end{pmatrix}$ is diagonally dominant. (T / **F**)

A.3 The QR decomposition can be used to solve $Ax = b$ even if the matrix A is not a square matrix. (**T** / F)

A.4 The Langrange interpolating polynomial for 5 points is identical with the Vandermonde intepolating polynomial over the same 5 points. (**T** / F)

A.5 The roots of the Chebyshev polynomials on the interval $[-1, 1]$ are always symmetric around 0. (**T** / F)

A.6 The Gauss-Seidel iterative method is essentially a fixed point iteration. (**T** / F)

A.7 Solving a least squares problem with the QR decomposition is better than solving the same least squares problem with the normal equations. (**T** / F)

A.8 The resulting system $Ax = b$ which we must solve while constructing a cubic spline is always diagonally dominant. (**T** / F)

A.9 A Gaussian quadrature with the form given below must have ADA= 6, (T / **F**)

$$\int_b^a f(x)\, dx \approx w_0 f(x_0) + w_1 f(x_1) + w_2 f(x_2).$$

.

Section B. (16 pts) Short explanations! Some solving or plotting may be needed. One to two sentences in some cases should be sufficient to explain your reasoning.

B.1 Write down the Taylor polynomial of degree 2 in order to approximate $f(x) = \exp(2 - x)$. What is $f(1)$ according to that Taylor polynomial?
Sol. Many choices for x_0 are possible. It is important that to choose a value for x_0 which would be close to the evaluation value of $x = 1$. Let's take $x_0 = 0$ but also $x_0 = 1$ or $x_0 = 2$ would work. A decimal value for x_0 which is close to 1 is fine but makes computations by hand harder.

$$f(x) \approx f(x_0) + \frac{f'(x_0)}{1!}(x - x_0)^1 + \frac{f''(x_0)}{2!}(x - x_0)^2 = e^2 - e^2 x + \frac{e^2 x^2}{2}.$$

As a result, $f(1) = e^2/2$

B.2 Suppose the condition number of a matrix A is 10^9. Let's say we multiply the matrix A with a small number $\epsilon = 0.1$. Does that reduce the condition number?
If yes then what is the new condition number. If not then why not.
Sol. The condition number does not change and it will definitely not get smaller (or bigger) if we multiply with a nonzero number. We can compute the condition number from a number of different formulas. One of the formulas presented in class used the 2-norms of A and A^{-1} which simplifies to the maximum and minimum eigenvalues of AA^T as follows,

$$\kappa(A) = ||A||_2 ||A^{-1}||_2 = \frac{\sqrt{\max \lambda_{A^T A}}}{\sqrt{\min \lambda_{A^T A}}}$$

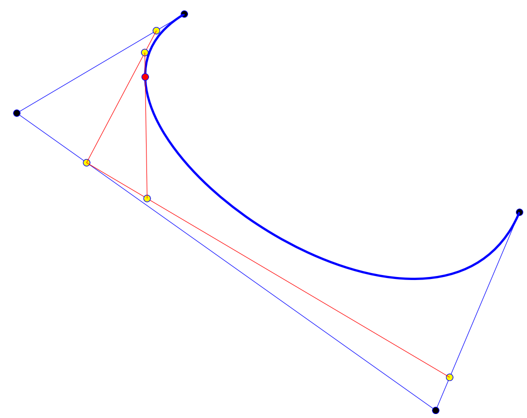
Note however that if we multiply a matrix A with a nonzero number ϵ the above equation will simply change to

$$\kappa(\epsilon A) = ||\epsilon A|| ||(\epsilon A)^{-1}|| = \frac{\sqrt{\epsilon^2 \max \lambda_{A^T A}}}{\sqrt{\epsilon^2 \min \lambda_{A^T A}}} = \frac{\sqrt{\max \lambda_{A^T A}}}{\sqrt{\min \lambda_{A^T A}}} = \kappa(A).$$

So the condition number does not change.

B.3 Is the Chebyshev intepolating polynomial for 5 points the same as the polynomial constructed by the Lagrange intepolating polynomial for the same 5 points?
Explain why yes or why not.
Sol. Note that the polynomials are both constructed over the same 5 points. Since Chebyshev uses the same methodoloty as Lagrange to construct the polynomial then both polynomials must be the same.

B.4 Draw the Bézier curve corresponding to the 4 points, $P_0 = (1, 1), P_1 = (0, 4), P_2 = (3, 3)$ and $P_3 = (2, 0)$ **by using de Castelju algorithm**.
Make sure to not only show the resulting Bézier curve but also to show how you constructed that curve by following de Castelju algorithm (construct at least 2 points of the Bézier curve). Nothing needs to be written. The solution should only be drawn.
Sol. Here is a figure of how 1 point would be constructed. Continue in the same way and show a couple more points....



B.5 Give **2** reasons for which a trigonometric interpolating polynomial would be preferred over any other interpolating method we learned in class?
Sol. The two reasons are:
1. Fast $O(n \log(n))$.
2. No system $Ax = b$ to solve for - no inverse is needed.

B.6 We showed in class that Gaussian quadratures can achieve high ADA thanks to a remarkable property of the Legendre polynomials. What is that property?
Sol. The Legendre polynomials of order n have the amazing property that when we multiply them with any other polynomial of degree less than $n-1$ and integrate it from -1 to 1 it will equal zero! In other words if $L_n(x)$ is a Legendre polynomial of degree n and $p_k(x)$ is any polynomial of degree k then,

$$\int_{-1}^{1} L_n(x)p_k(x)\, dx = 0 \quad \text{for any } k < n.$$

B.7 Consider the following Newton-Cotes quadrature formula,

$$\int_a^b f(x)\, dx \approx w_0 f(x_0) + w_1 f(x_1) + w_2 f(x_2).$$

What is the ADA for this formula?
Sol. For $n + 1 = 3$ points implies that $n = 2$ which is even and therefore the ADA$= 3$ (see lecture notes).

B.8 Suppose you wish to compute the condition number of the matrix A involved in solving,

$$\begin{aligned} 0.77x_1 + x_2 &= 14.25 \\ 1.2x_1 + 1.7x_2 &= 20 \end{aligned}$$

Using methods we discussed in class describe the exact methodology you would use to compute the condition number in the $||\cdot||_2$ norm. Do not compute anything!

Sol. We first try to estimate the largest (in absolute value) eigenvalue using the Power iteration without QR for the matrix A^TA where A corresponds to the matrix for the system above (note that A is not a symmetric matrix). Once that largest eigenvalue is found then we use the method of Deflation to create a new matrix B. We now apply the Power iteration without QR again to obtain an estimate of the next eigenvalue which in this example would be the smallest eigenvalue. Finally to estimate the condition number we simply compute the fraction of

$$\kappa(A) = \frac{\sqrt{max(\lambda_{A^TA})}}{\sqrt{min(\lambda_{A^TA})}}$$

where the $\max(\lambda_{A^TA})$ and $\min(\lambda_{A^TA})$ are the largest and smallest eigenvalues of A^TA.

Section C. Problems in the following pages must be worked out completely to receive any credit.
Justify all your answers and write down all steps.
Correct answers without justification will receive a 0.

1. **(4pts)** Consider the Matlab code below,

Algorithm 0.1 Matlab Algorithm

$x(1) = 1/2$
for $k = 1 : 10$
  $x(k+1) = cos(x(k))/2$
end

   (a) Which numerical method is implemented in the code? What does the $x(11)$ approximate?
   Sol. The algorithm implements the fixed point iteration and solves $x = 1/2\cos(x)$**.**

   (b) Prove that the method converges, as k tends to infinity, for any choise of $x(1)$ in the interval $[0,1]$.
   Sol. Find the derivative and show that in absolute value it is less than 1 in the given interval: $g(x) = \cos(x)/2)$**. So** $g'(x) = -\sin(x)/2$ **which is less than 1, in absolute value, for any** x **in the interval** $[0,1]$

2. **(4pts)** Construct the least-squares solution which fits the curve $f(t) = at^b e^{c(t-1)}$ for the data points $(0,0), (1,2), (2,4/e), (3,6/(e^2))$. Only set-up the corresponding linear system $Ax = b$ for this problem. **Do not solve it!**
Sol. Note: $\log(0)$ **does not exist and that's why we do not use it in the resulting system.**
We first linearize the function by taking the logarithm (base e**) of both sides,**

$$\log(f) = \log(a) + b\log(t) + c(t-1).$$

Now we substitute the provided data for t **and** f **and obtain the following system,**

$$\begin{pmatrix} \log(2) \\ \log(4/e) \\ \log(6/e^2) \end{pmatrix} = \begin{pmatrix} 1 & \log(1) & 0 \\ 1 & \log(2) & 1 \\ 1 & \log(3) & 2 \end{pmatrix} \begin{pmatrix} \log(a) \\ b \\ c \end{pmatrix}.$$

3. **(4pts)**

   (a) Compute the **Legendre** polynomial of order 4.
   Sol. The Legendre polynomials of order n **are defined via,**

$$P_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}(x^2-1)^n.$$

As a result the Legendre polynomial of order 4 is given by,

$$\frac{1}{8}(35x^4 - 30x^2 + 3).$$

(b) What exactly are the roots of the Legendre polynomial used for? (do not compute anything-only answer the question.)
Sol. The roots of the Legendre polynomial are used as the nodes upon which Gaussian quadrature methods are to be evaluated. (These nodes are responsible for the fact that Gaussian quadratues can achieve much higher ADA when compared to Newton-Cotes formulas).

4. **(4pts)**

(a) Compute all the 5-th roots of unity. Show all your work (no graphical solutions allowed).
Sol. The 5 roots of unity can be computed by solving the following equation for the complex number $1 = e^{i2k\pi}$ for $k = 0, 1, 2, 3, 4$,

$$z^5 = 1 \quad \textbf{which we re-write as} \quad z = \sqrt[5]{1} = e^{i2k\pi/5}.$$

We use Euler's formula and a sub index k to denote each root z via

$$z_k = \cos(2k\pi/5) + i\sin(2k\pi/5) \quad \textbf{for } k = 0, 1, 2, 3, 4.$$

As a result for each value of k we obtain the 5 roots of unity,

$$\begin{aligned} r_0 &= 1 \\ r_1 &= \cos(2\pi/5) + i\sin(2\pi/5) \\ r_2 &= \cos(4\pi/5) + i\sin(4\pi/5) \\ r_3 &= \cos(6\pi/5) + i\sin(6\pi/5) \\ r_4 &= \cos(8\pi/5) + i\sin(8\pi/5). \end{aligned}$$

(b) Which of those are the primitive roots?
Sol. Here r_1 is one of the primitive roots of unity since we can get all others from it.

5. **(4pts)** Determine the value of a, b and c that makes the function $f(x)$ a cubic spline,

$$f(x) = \begin{cases} x^3 & \text{for } x \in [0,1], \\ \frac{1}{2}(x-1)^3 + a(x-1)^2 + b(x-1) + c & \text{for } x \in [1,3]. \end{cases}$$

Sol. Note that we need continuity of the function, the derivative and the second derivative. In that respect, each of those conditions, gives the following relationships,

$$\begin{aligned} f(1) &= 1^3 = 1 = c, \quad \textbf{So } c = 1 \\ f'(1) &= 3 \cdot 1^2 = b, \quad \textbf{So } b = 3 \\ f''(1) &= 6 \cdot 1 = 2a, \quad \textbf{So } a = 3 \end{aligned}$$

6. **(3pts)** Given the ODE and corresponding initial conditions below,

$$\begin{cases} y''' &= y - y' + t, \\ y''(0) &= -1, \\ y'(0) &= 0, \\ y(0) &= 1, \end{cases}$$

reduce it to a system of 1st order ODEs - make sure to also include their initial conditions.

Sol. We follow the lecture notes to create a system of 3 equations in 3 variables u_1, u_2 and u_3. This gives the following system,

$$\begin{cases} u_1' &= u_2 \\ u_2' &= u_3 \\ u_3' &= u_1 - u_2 + t, \end{cases}$$

with initial conditions $u_1(0) = 1, u_2(0) = 0$ and $u_3(0) = -1$.

7. **(5pts)** Consider the ODE,

$$\begin{cases} y' &= sin(y) + t^2, \\ y(0) &= \pi/2. \end{cases}$$

(a) Write down the **implicit Euler** method for **this** ODE.
Sol. The implicit Euler method is $w_{n+1} = w_n + hf(t_{n+1}, w_{n+1})$.
The implicit Euler method however for this ODE is,

$$\begin{aligned} w_{n+1} &= w_n + h\left(\sin(w_{n+1}) + t_{n+1}^2\right) \\ t_{n+1} &= t_n + h. \end{aligned}$$

(b) If you use the fixed point iteration $x = g(x)$ to solve for w_{n+1} what exactly is the function $g(x)$ for this problem? Make sure to indicate which parameters will stay constant and which will be changing during the fixed point iteration.
Sol. The function $g(x)$ for the fixed point iteration is, $g(x) = w_n + h\left(\sin(x) + t_{n+1}^2\right)$. Here w_n, h and t_{n+1} will be constants during the iteration. The variable x will be the only one changing during the iteration.

(c) If you use Newton's iteration instead to solve for w_{n+1} then what exactly is the function $f(x)$? Make sure to indicate which parameters will stay constant and which will be changing during the Newton iteration.
Sol. If we use Newton's iteration instead to solve for w_{n+1} then the function $f(x) = x - w_n - h\left(\sin(x) + t_{n+1}^2\right)$. Here w_n, h and t_{n+1} will be constants during the iteration. The variable x will be the only one changing during the iteration.

(d) Now use the **explicit Euler** method and perform a single iteration with $h = 1$ and compute the value for w_1 (no need to simplify the resulting expression).

Sol. Using the explicit Euler method we must iterate the following formula,

$$
\begin{aligned}
w_{n+1} &= w_n + h\left(\sin(w_n) + t_n^2\right)\\
t_{n+1} &= t_n + h.
\end{aligned}
$$

Note that $w_0 = \pi/2$ **and** $t_0 = 0$**. So iterating the above for** $h = 1$ **gives the solution** $w_1 = 1 + 1 \cdot (\sin(\pi/2) + 0^2) = 1 + \sin(\pi/2) = 1$**. So** $w_1 = 1$**.**

Good Luck!