

ICSI 401 – Numerical Methods

Instructor: Abram Magner

Date: Fall 2020

Please answer the following questions in complete sentences, showing all work (including code and output of programs, when applicable). I prefer that you type your solutions (e.g., using LaTeX, with Overleaf, TeXworks, etc., or Word), but will accept handwritten notes. If the grader cannot read your handwriting, then they cannot award you points.

Certain problems require you to write Matlab code. You should include in your submission a printout of all Matlab code that you write, in addition to the output that you get when you run it.

Due date: Wednesday, 11/18/2020, 11:59 p.m., on Blackboard.

4.1 Partial pivoting

In this problem, you will review partial pivoting and the reasoning behind it.

1. Circle the best explanation for the use of partial pivoting from the choices below.
 - (a) Partial pivoting is used in Gaussian elimination to mitigate the effects of rounding errors in the matrix entries, resulting in an algorithm that is usually backward stable.
 - (b) Partial pivoting is not used in Gaussian elimination at all, but is instead used to solve linear systems via Cramer's rule.
 - (c) Partial pivoting is used to transform the input to Gaussian elimination so as to speed up the solution of linear systems.
2. Show the steps of Gaussian elimination with partial pivoting for the following matrix:

$$\begin{pmatrix} 10^{-5} & 1 & 0 \\ 2 & -3 & 1 \\ -1 & 1 & 1 \end{pmatrix} \quad (4.1)$$

Make sure that you clearly indicate in each step what operation you are performing (e.g., which rows are swapped, which multiples of rows are added to which other rows) and the result of that operation. You should end up with an upper triangular matrix.

4.2 Least squares and curve fitting

This problem will give you practice in using least squares to fit polynomials to data. Consider the following dataset in Matlab, consisting of pairs of points (x, y) :

```
x = [-3, -2, -1, 0, 1, 2, 3]';
y = [-12.529999999999999, -3.02, 0.49, 1.00, 1.51, 5.02, 14.529999999999999]';
```

Suppose that you want to find a least squares fit of a degree-3 polynomial to this data. I.e., you want to determine coefficients c_0, c_1, c_2, c_3 such that $c_3x^3 + c_2x^2 + c_1x + c_0$ is a least-squares fit to the data. We will collect the coefficients into a vector $\vec{c}_* = [c_0, c_1, c_2, c_3]^T$.

1. Using Matlab, compute the Vandermonde matrix A associated with this problem and explain how you got it. Remember that least squares requires us to solve

$$\vec{c}_* = \operatorname{argmin}_{\vec{c}} \|A\vec{c} - \vec{y}\|_2^2. \quad (4.2)$$

Hint: In Matlab, if you have some number of column vectors a, b, c, \dots of the same length, then you can put them into a matrix as follows:

```
A = [a , b , c];
```

Furthermore, you can generate a column vector of all ones with a given length l by writing

```
ones(l)
```

Finally, given a vector x , you can raise each of its elements to a given power a by writing

```
x.^a
```

2. In Matlab, use the A that you computed and the **normal equations** approach to the least squares problem to compute the vector of coefficients \vec{c} .
3. In Matlab, use the *norm* function to compute the error $\|A\vec{c} - \vec{y}\|_2^2$.

For this problem, you are expected to turn in all Matlab code that you've written, and clearly indicate your final answers.

4.3 Eigenthings and the power method

1. Using Matlab, use the power method to calculate unit eigenvectors corresponding to the dominant two eigenvalues of the matrix

$$\begin{pmatrix} 6 & 2 & -1 \\ 2 & 5 & 1 \\ -1 & 1 & 4 \end{pmatrix} \quad (4.3)$$

In particular, you should run the power method for 50 iterations.

As usual, include your Matlab code and output. Clearly indicate the two eigenvectors that are your final answer.

Hint: Think about how you might check your answer to make sure it's accurate. You know that an eigenvector v must satisfy $Av = \lambda v$ for some λ , so one idea is to compute $w = Av$ and make sure that $w_1/v_1 = w_2/v_2 = w_3/v_3$.

4.4 Polynomial interpolation, Chebyshev points

1. In general, if we are given points $(x_0, y_0), \dots, (x_n, y_n)$, what is the minimum number d such that a unique polynomial with degree $\leq d$ and passing through these points is guaranteed to exist?
2. Let $x_0 = -1, x_1 = 1, x_2 = 2$, and let $y_0 = 1, y_1 = 2, y_2 = 3$. Write down the Lagrange interpolating polynomial that passes through the points $(x_0, y_0), (x_1, y_1), (x_2, y_2)$.
3. For the points in the previous problem, fill in the following divided difference table:

$f[x_0]$		
$f[x_1]$	$f[x_0, x_1]$	
$f[x_2]$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$

In other words, what is expected is that you compute $f[x_0], f[x_1], f[x_0, x_1]$, etc.

Hint: Remember that the point of divided differences is that they can be computed recursively. See the textbook for the definition and more information.

4. Using the divided difference table above, write down the Newton interpolating polynomial for these points.

Hint: Your answer should be of the form $a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1)$. So what you need to do is calculate a_0, a_1, a_2 from the divided differences table.

Also, note that the polynomial that you get from Newton interpolation should be the same as the polynomial that you get from Lagrange interpolation. So you can check your answer by plugging in some points into each polynomial and making sure that you get the same result from each.

4.5 Piecewise polynomial interpolation

1. Do Exercise 8.8.9 in the book (on piecewise polynomial interpolation).

Hint: The constraints that $P(x)$ and $P'(x)$ be continuous at $x = 1$ mean that we must have $P_1(1) = P_2(1)$ and $P'_1(1) = P'_2(1)$. This is because, intuitively, continuous functions are those functions that don't have any gaps or jumps.