Computational Statistics HW1

Liam Carpenter

October 8 2021

Problem 1

The use of newtons method depends on the existence of a taylor series for a given function. So for any differentiable function in the complex plane we can write a first order taylor series that will be the best linear approximation of the function. In the case of our function

$$f(z) = z^2 - 2z + 2, z \in C$$

We can write a first order Taylor series approximation analogous to the real case around a point z_0,

$$f(z) \approx f(z_0) = f(z_0) + f'(z_0)(z - z_0)$$

Which we can then set to zero and solve for the next iterations. Ie.

$$f(z_0) + f'(z_0)(z_1 - z_0) = 0$$

$$\implies z_1 = z_0 - \frac{f(z_0)}{f'(z_0)}$$

So really the we can write a newtons method to find complex roots by just considering complex variables as our inputs into the system.

Problem 2

1. What is the cost of computing $(H^{k+1})^{-1}$ using the woodbury matrix identity? The solution to the inverse using the Woodbury matrix identity is for our rank two update is

$$(H^{(k+1)})^{-1} = (H^k)^{-1} + \frac{ss^T + (g.T(H^{(k)})^{-1}g)(ss^T)}{(s^Tg)^2} - \frac{(H^k)^{-1}gs^T + sg^T(H^k)^{-1}}{s^Tg}$$

   This quanitity can be computed using strictly Matrix Vector products and Matrix Matrix addition, making the computational cost of forming the inverse Hessian of a rank two update $\mathcal{O}(n^2)$.

2. If we instead wish to simply apply the inverse Hessian we can cut down on or cost a bit.

$$(H^{(k+1)})^{-1}\nabla f(x^{k+1}) = ((H^k)^{-1} + \frac{ss^T + (g.T(H^{(k)})^{-1}g)(ss^T)}{(s^Tg)^2} - \frac{(H^k)^{-1}gs^T + sg^T(H^k)^{-1}}{s^Tg})\nabla f(x^{(k+1)})$$

$$= ((H^k)^{-1}\nabla f(x^{(k+1)}) + \frac{ss^T + (g.T(H^{(k)})^{-1}g)(ss^T)}{(s^Tg)^2}\nabla f(x^{(k+1)}) - \frac{(H^k)^{-1}gs^T + sg^T(H^k)^{-1}}{s^Tg})\nabla f(x^{(k+1)})$$

The total computational complexity of this is still $\mathcal{O}(n^2)$ but we manage to do away with the additional $\mathcal{O}(n^2)$ matrix additions and reduce them to vector additions which are $\mathcal{O}(n)$. Therefore we are able to reduce some of the computation needed to compute our desired product.

3. Assuming that we have $(H^k)^{-1}$ the most efficient way between the two of these methods would be to do the application alone. Additionally since the Hessian is symmetric positive definite, instead of doing update on the full hessian we can compute the updates on Cholesky factors of the Hessian.

Problem 3

1. (a) Show $\|\mathbf{u}\|_\infty \leq \|\mathbf{u}\|_2$. Let $u_k, 1 \leq k \leq n$ be the maximum absolute value element of the vector $\mathbf{u}$. Expanding both sides of the inequality and taking the square.

$$u_k^2 \leq u_1^2 + u_2^2 + \cdots + u_n^2$$
$$\implies 0 \leq u_1^2 + u_2^2 + \cdots + u_{k-1}^2 + u_{k+1}^2 + \cdots + u_n^2$$

Which is obviously true.

(b) Show $\|\mathbf{u}\|_2^2 \leq \|\mathbf{u}\|_1\|\mathbf{u}\|_\infty$ Expanding out both sides again we have

$$\sum_{i=1}^{n} u_i^2 \leq |u_k| \sum_{i=1}^{k} |u_i|$$

Since u_k is the maximal element we have the inequality $u_i^2 \leq |u_k||u_i|, \forall, 1 \leq i \leq n$. This proves the inequality as each element on the RHS is greater than or equal to each element on the LHS.

Taking the vector $\mathbf{u} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$ We can see that $\|\mathbf{u}\|_\infty = 2 \leq \sqrt{5} = \|\mathbf{u}\|_2$. Further $\|\mathbf{u}\|_2^2 = 5 \leq 6 = 2(2+1) = \|\mathbf{u}\|_1\|\mathbf{u}\|_\infty$

(c) We have already proven $\|\mathbf{u}\|_\infty \leq \|\mathbf{u}\|_2$ so know we just need to show that $\|\mathbf{u}\|_2 \leq \|\mathbf{u}\|_1$ which is easily seen by taking the square of both sides.

$$\sum_{i=1}^{k} u_i^2 \leq (\sum_{i=1}^{k} |u_i|)^2$$

$$RHS = \sum_{i=1}^{n} u_i^2 + 2\sum_{i\neq j}^{n} |u_i||u_j|$$

So obviously $\|\mathbf{u}\|_\infty \leq \|\mathbf{u}\|_2 \leq \|\mathbf{u}\|_1$

(d) Again note that we are considering u_k to be the maximal element of $\mathbf{u}$. We can prove this inequality by squaring

$$\sum_{i=1}^{n} |u_i|^2 \leq n|u_k|^2$$

Since this is true the square root of this inequality will also be true.

2. (a) The first identity is easy enough using the knowledge that the infinity norm is equivalent to the maximum row sum of a matrix and the L-2 norm is the largest singular value. With this in mind we take the SVD of the matrix A in the infinity norm.

$$\|\mathbf{A}\|_\infty \leq \sqrt{n}\|\mathbf{A}\|_2$$
$$\|\mathbf{A}\|_\infty = \|\mathbf{U}\Sigma\mathbf{V}^T\|_\infty \leq \|\mathbf{U}\|_\infty\|\Sigma\|_\infty\|\mathbf{V}^T\|_\infty$$

Since both U and V are orthonormal we know that all of their rows and columns sum to 1 therefore the infinity norm of these matrices are each 1. Further Σ is a diagonal matrix of singular values of A so the infinity norm is just the largest singular value. Denoting σ_{max} as the largest singular value we can see

$$\|\mathbf{A}\|_\infty \leq \|\Sigma\|_\infty = \sigma_{max} \leq \sqrt{n}\sigma_{max} = \sqrt{n}\|\mathbf{A}\|_2$$

(b)

$$\|\mathbf{A}\|_2 = sup\frac{\|\mathbf{Au}\|_2}{\|\mathbf{u}\|_2} \leq sup\frac{\|\mathbf{Au}\|_2}{\|\mathbf{u}\|_\infty}$$

Since we have that $\|\mathbf{u}\|_\infty \leq \|\mathbf{u}\|_2$. Finally we note that $\mathbf{Au}$ is just a vector $\in R^m$ and therefore our previous inequality $\|\mathbf{u}\|_2 \leq \sqrt{m}\|\mathbf{u}\|_\infty$ holds.

$$sup\frac{\|\mathbf{Au}\|_2}{\|\mathbf{u}\|_\infty} \leq \sqrt{m}sup\frac{\|\mathbf{Au}\|}{\|\mathbf{u}} = \sqrt{m}\|\mathbf{A}\|_\infty$$

Let's consider the matrix $\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}$

$$\|\mathbf{A}\|_\infty = 3 \leq 4.13 = \sqrt{2} \times 2.92 = \sqrt{m}\|A\|_2$$

Similarly

$$\|\mathbf{A}\|_2 = 2.92 \leq 3 \times \sqrt{2} = \sqrt{m}\|\mathbf{A}\|$$

Problem 4

Roots of P_{16}.

Root Number	Root Values
0	-0.990352
1	-0.948187
2	-0.871826
3	-0.763149
4	-0.625807
5	-0.464796
6	-0.286142
7	-0.096606
8	0.096606
9	0.286142
10	0.464796
11	0.625807
12	0.763149
13	0.871826
14	0.948187
15	0.990352

Problem 5

To show that the $f(\mathbf{x})$ is convex we will show that $\mathbf{A}$ is positive semi-definite and therefore because the function take a quadratic form has to be convex. To do this we will show that $\forall \mathbf{x}, \mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$ Take arbitrary $\mathbf{x} \in R^n$ expanding out the product $\mathbf{x}^T\mathbf{A}\mathbf{x}$ we get

$$2(\sum_{i=1}^{n} x_i^2 - \sum_{i=2}^{n} x_{i-1}x_i)$$

Further we have the inequality $\frac{x_i^2+x_j^2}{2} \geq x_i x_j$. Re-writing each $x_{i-1}^2 + x_i^2 = \frac{x_{i-1}^2+x_i^2}{2} + \frac{x_i^2+x_{i+1}^2}{2}$,

$$\sum_{i=1}^{n} x_i^2 = \frac{x_n + x_1}{2} + \sum_{i=2}^{n} \frac{x_{i-1} + x_i}{2} \geq \sum_{i=2}^{n} x_{i-1}x_i$$

This shows us that $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$ and therefore our matrix is positive semi-definite and further our function $f(\mathbf{x})$ is convex.

Table of iterates. We are doing n=10 so it was pretty infeasible to actually show the entirety of the x values, but I did my best.

Iteration	Function Values	Absolute Precision
1	143.000000	2.353720e+01
2	2712.683505	1.137232e+02
3	3154.008103	4.842761e+01
4	2822.399739	6.928676e+01
5	2096.243974	4.355153e+01
6	2955.149714	7.639611e+01
7	1377.754973	4.082253e+01
8	1340.188542	4.443870e+00
9	1355.965803	1.039308e+00
10	1354.335186	8.273726e-02
11	1354.497453	8.812360e-03
12	1354.500311	2.904346e-04
13	1354.499998	2.050737e-05
14	1354.499999	2.039661e-07
15	1354.500001	2.149092e-07
16	1354.500000	1.498150e-07
17	1354.500000	1.238696e-07
18	1354.500000	1.115655e-07
19	1354.500000	3.364108e-08
20	1354.500000	1.297898e-08
21	1354.500000	1.693585e-08
22	1354.500000	1.175222e-08
23	1354.500000	1.886389e-10

Iteration	X(Rounded to two Sinificant Figures)
1	[-8.0, -6.0, -6.0, -6.0, -6.0, -6.0, -6.0, -6.0...
2	[-20.9, -50.5, -44.0, -44.0, -44.0, -44.0, -44....
3	[-22.3, -38.0, -66.2, -60.1, -60.1, -60.1, -60....
4	[-15.8, -15.3, -24.5, -56.7, -50.2, -50.2, -56....
5	[-14.3, -25.2, -24.4, -27.8, -54.5, -54.5, -27....
6	[-20.1, -46.7, -62.7, -58.2, -49.0, -49.0, -58....
7	[-17.6, -29.7, -41.3, -50.7, -53.3, -53.3, -50....
8	[-17.4, -31.6, -41.7, -48.4, -52.3, -52.3, -48....
9	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
10	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
11	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
12	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
13	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
14	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
15	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
16	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
17	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
18	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
19	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
20	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
21	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
22	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....
23	[-17.5, -31.5, -42.0, -49.0, -52.5, -52.5, -49....

Problem 6

The code is forming a matrix and then forming a random vector. These two quantities are multiplied to get a b vector. The system is then treated as though x is unknown and the system is solved by computing $\mathbf{A}^{-1}\mathbf{b}$. The final computation is evaluating the norm of the difference between $\mathbf{x} - \mathbf{y}$. We see that there is some tangible error even though mathematically the two solutions should be the same. This error can be attributed to floating point arithmetic and the condition number on the matrix A. Matlab gives us that that condition number is $\kappa(A) = 6.5891 \times 10^{10}$. The significant digits lost in in solving the linear system can be approximated by $-log_{10}(\kappa(A) \times \epsilon_{machine}) = 4.8347 \approx 5$. The Error in our Matlab code is $2.8077 \times 10^{-}6$ which agrees with our estimate that there will be numerical inaccuracies past the 5th digit of precision.