DSCI 320: Optimization Methods in Data Science

Homework assignment 2 – due Friday 10/04/2019

Problem 1 (Convergence order). Determine the order of convergence and the asymptotic error constant for the following sequences:

(a) $a_k = 5.0625, 2.25, 1, \frac{4}{9}, \frac{16}{81}$

(b) $b_k = 2.718, 2.175, 1.740, 1.392, 1.113, 0.8907$

(c) $c_k = 0.318, 0.180, 0.0761, 0.021, 3.04 \cdot 10^{-3}, 1.68 \cdot 10^{-4}, 2.17 \cdot 10^{-6}$. You would generally do this by assuming a $a_{k+1} = Ca_k^s$ behavior of the sequence and seeing what values for C and s make this sequence as it is. You can then try to "guess" C and s from subsequent values of the sequence, or use a more systematic approach. **(5 points)**

Problem 2 (Steepest descent iteration). For badly conditioned problems, the steepest descent algorithm takes exceedingly long. Let us verify this claim:

Consider a matrix and vector A, b

$$A = \begin{pmatrix} 10 & 0 \\ 0 & 1 \end{pmatrix}, \qquad b = \begin{pmatrix} 10 & 0 \end{pmatrix}.$$

and an objective function

$$f(x) = \frac{1}{2} x^T A x - x^T b.$$

The minimum of this function lies at $x^* = (1, 0)$. Generate graphs that show the surface and contours of the function $f(x)$.

Next consider the steepest descent iteration. Start from $x_0 = (2, 10)$. Perform 100 iterations, where in each iteration you compute

$$p_k = -\nabla f(x_k) = b - Ax_k, \qquad \alpha_k = \frac{p_k^T p_k}{p_k^T A p_k},$$

and then set $x_{k+1} := x_k + \alpha_k p_k$. The formula for α_k implements a particular step length rule that is appropriate for this kind of problem and when using the steepest descent method.

Plot the iterates $x_k = (x_{k1}, x_{k2})$ in a 2-dimensional plot and connect them by lines to see their convergence towards x^*.

How many iterations do you need to achieve an accuracy of $\|x_k - x^*\| \leq 10^{-4}$? Repeat the experiment where a_{11} and b_1 both have the values 1, 10, 100, 1000, 10000 (all other elements of A and b unchanged), and starting from $x_0 = (2, a_{11})$. Create a table with the condition number of these matrices and how many iterations it takes to achieve above accuracy. **(30 points)**

Problem 3 (Newton's method). Repeat the previous problem, but instead of using the steepest descent algorithm use Newton's method with

$$p_k = -[\nabla^2 f(x_k)]^{-1} \nabla f(x_k), \qquad \alpha_k = 1.$$

Note that here, $\nabla^2 f(x_k) = A$ and as before, $\nabla f(x_k) = Ax_k - b$.

Explain your observations and compare what you see with what you saw for the gradient method. **(15 points)**

Problem 4 (Slow convergence of Newton's method). While generally considered very fast, there are cases where even Newton's method makes only very slow progress. Examine the problem of finding the minimum of the one-dimensional function $f(x) = x^{30}$, starting at an arbitrary poing x_0. The minimum, of course, lies at $x^* = 0$. Write down the equation for the search direction p_k, given x_k. In the following, assume that we choose a step length of $\alpha_k = 1$ in every iteration.

For the concrete choice $x_0 = 20$, write a little program that finds the minimum using Newton's method. Plot the distance $|x_k - x^*|$ as a function of the iteration number k. How many function and gradient evaluations do you need to achieve an accuracy of $|x_k - x^*| < 10^{-4}$? What is the convergence order you observe? **(20 points)**

Problem 5 (Radius of convergence for Newton's method). You have seen in class that Newton's method (with step length $\alpha_k = 1$) can only be proven to converge if we have a starting point x_0 that is close enough to the solution x^*. One reason may be that far enough away from the minimum x^* the function $f(x)$ may not be convex any more. Explain in words for a function $f(x)$ of a single variable $x \in \mathbb{R}^1$ what would go wrong if we started in an area where $f(x)$ is not convex (i.e. where $f''(x) < 0$). In particular, think about how Newton's method chose its search direction. If you want a concrete example to explain things with, use the function

$$f(x) = -\frac{1}{1+x^2}$$

whose minimum is at $x^* = 0$ but whose second derivative is positive only in an interval around the origin (i.e., since we are in 1d, whose curvature is positive only around the origin, and which is therefore convex only around the origin).

You can illustrate your explanation with numerical results that show what Newton's method does if, for example you start at $x = 0.1, 0.5, 1, 2, 5, \ldots$. **(15 points)**

Problem 6 (Radius of convergence for Newton's method). This problem illustrates the need for a strategy to determine the step length α_k when using Newton's method. If you start close enough to the solution, Newton's method also converges if you choose $\alpha_k = 1$, but if you're still far away from the solution, one needs to tread more carefully.

The case discussed in the previous problem is not the only one where Newton's method may not converge. Consider

$$f(x) = x\ \arctan x - \frac{1}{2}\ln(1+x^2).$$

This function is convex everywhere since $f''(x) = \frac{1}{1+x^2} > 0$. Yet, Newton's method (with step length $\alpha_k = 1$) only converges if started within an interval $[-r, r]$ around the minimum $x^* = 0$. Determine numerically the radius of convergence r for this problem. What happens if you start with $x_0 = r$? What if $x_0 > r$?

(15 points)

Bonus problem (The power of looking at problems differently). Given data points $\{t_i, y_i\}$ there were different ways to fit a line $y(t) = at + b$ through them. Among them were the least sum of squares (or, in short, the "least squares" method), the least sum of absolute values, and the least maximal value objective function. In last week's homework, you had seen that the objective function that corresponds to the latter two was non-smooth. On the other hand, on the slides that were shown during the first two classes, you had seen a trick that can reformulate the least-absolute-values problem from a non-smooth unconstrained one into a constrained problem in which both objective function and constraints were linear – i.e. a problem that is much simpler to solve.

Can you find a way in which the least-maximal-value problem that corresponds to the objective function $f(x) = \max_i |y_i - y(t_i)|$ can be reformulated in a similar way, yielding a linear problem with linear inequalities? If so, compare the number of additional variables and the number of inequalities needed to reformulate the maximal difference and sum of differences problems.

(10 bonus points)

If you have comments on the way I teach – in particular suggestions how I can do things better, if I should do more or less examples, powerpoint slides vs whiteboard, etc – or on other things you would like to critique, feel free to hand those in with your homework as well. I want to make this as good a class as possible, and all comments are certainly much appreciated!