Math 661 — Homework
Optimization — September 28, 2020

# Basic Newton Optimization

1. Write MATLAB code to minimize an objective function of $n$ variables, using a Newton approach with backtracking and the following features: Here $\epsilon$, $m$, $\mu$ are as on p. 377 of the text and should be set in advance.

   (a) Specify the objective function, its gradient, and its Hessian as functions.

   (b) At each iteration, compute the Newton direction. If it is not a sufficient descent direction, replace it by the gradient descent direction.

   (c) If the search direction is not gradient related, rescale it so it is.

   (d) Using backtracking, with initial choice 1, choose a step size giving sufficient decrease.

   (e) Stop your iteration when

   $$\frac{||\nabla f(x_i)||}{1 + |f(x_i)|} < \delta,$$

   for some $\delta$ set in advance.

   (f) Print out your estimate of a local minimizer, the number of iterations, and the values of $f$ and $\nabla f$ at your estimate.

2. Test your code on problem 3.3 on p. 369, with $\delta = 10^{-12}$, $\epsilon = m = \mu = 0.1$ to compare the performance with your previous work on that problem. How do the number of iterations differ with starting point $x_0 = (1, 1)$? Can you find any starting point where the new code requires substantially fewer iterations? substantially more iterations? Look at a plot of the function and explain why you would or would not expect the two versions of the code to behave similarly.

3. Consider the objective function, in which two coefficients have been changed from the previous problem.

   $$f(x, y) = 5x^4 + 6y^4 - 40x^2 + 2xy - 20y^2 + 15x - 7y + 13.$$

   Use your new code to find all local minima. Find interesting initial points where your new and old codes behave differently. Discuss why. You may find graphing the function helpful, but be aware that for real-world problems that is usually not possible.