MATH/AMATH 516, Spring 2009 Homework 1 Due April 15 at 10:30 AM

The page/equation numbers are from *Nonlinear Programming*, 2nd edition, 1999. They may differ for later edition.

You are allowed to discuss the problems with each other, but you must write up your answers on your own. Feel free to ask for a hint if you get stuck. [The page/equation numbers are from *Nonlinear Programming*, 2nd edition, 1999.]

The answers to the two * problems are to be turned in jointly with your (randomly chosen) partner.

#0. Read Appendix A on background materials.

#1. [Gradient, Hessian, convexity.] Optimization is often used for input-output modeling from given sample inputs $u^i \in \Re^n$ and corresponding outputs $y_i \in \Re$, $i = 1, ..., N$.

a) In a least square model, we assume the output is a linear function of the input $y = x^T u$, and the n unknown coefficients $x \in \Re^n$ are determined by minimizing the sum of the squared output errors

$$f(x) = \sum_{i=1}^{N} (x^T u^i - y_i)^2.$$

Find ∇f and $\nabla^2 f$. Express your answer in terms of $y = (y_1, \dots, y_N)^T$ and U, the $N \times n$ matrix whose ith row is $(u^i)^T$. Is f convex? Give a rigorous argument for your answer. Find a closed form expression for the unique global minimum of f when U has rank n. (Note: In a neural network model, we assume $y = \phi(x^T u)$, where $\phi(t) = 1/(1 + e^{-ct})$ with $c > 0$. Think of x as synaptic weights and ϕ models the activation function. The resulting error function is nonconvex, however.)

b) In a logistic regression model, we minimize the function

$$f(x) = \sum_{i=1}^{N} \ln(1 + e^{x^T u^i}) - y_i x^T u^i.$$

Rewrite f in the form $g(Ux)$, for some $g : \Re^N \to \Re$, with U defined as in (a). Find ∇f and $\nabla^2 f$. Express your answer in terms of ∇g, $\nabla^2 g$, and U. Is g convex? Is f convex? Give a rigorous argument for your answer.

#2. [Global convergence of gradient descent method.] Exercise 1.2.16, do part (a) only. (Here f is assumed to be continuously differentiable. The two criteria are sometimes called the "strong Wolfe condition".) You can replace the condition of d^k being gradient-related by the condition from lectures, i.e., ${g^k}^T d^k \le -c_1 \|g^k\|^2$ and $\|d^k\| \le c_2 \|g^k\|$ for all k $(0 < c_1 \le c_2)$.

#3.* [Growth estimate.] Exercise 1.2.10. (Hint: To show the first inequality, use the Fundamental Theorem of Calculus: $\nabla f(x) - \nabla f(x^*) = \int_0^1 \nabla^2 f(x^* + t(x - x^*))(x - x^*) dt$ and the Cauchy-Schwarz inequality $u^T v \le \|u\| \|v\|$. To show the second inequality, use the 2nd-order Taylor expansion of $f(x^*)$ at x, e.g., Proposition A.23(b).) This growth property is key to the linear convergence of gradient descent methods.

#4. [Computation problem.] On the class webpage is posted a Matlab program that implements the steepest descent method using the Armijo stepsize rule. The function and gradient routines for the Rosenbrock example $f(x_1, x_2) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2$ are also shown. (To run this, save the program and the two routines in three files named steepdesc.m, func.m, grad.m and, assuming your computer system has Matlab, start Matlab and type steepdesc) By modifying this program or writing from scratch in your favorite computer language, implement a method that (i) uses the Newton direction when $\nabla^2 f(x^k)$ is positive definite and otherwise uses the steepest descent direction, (ii) uses the Armijo stepsize rule. Run your program on the above Rosenbrock example with initial point $x^0 = (-1.2\ , 1)'$. Also, run your program on the Powell example $f(x_1, ..., x_4) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$ with initial point $x^0 = (3, -1, 0, 1)'$.

How does your program compare with steepest descent method in terms of (i) number of iterations, (ii) total number of function evaluations, (iii) total number of gradient evaluations? If function and gradient are expensive to evaluate, which method would you choose? If solving an $n \times n$ system of linear equations is expensive, which method would you choose?

#5.* [Application.] Two points, a and b, are connected by a chain consisting of n mass points joined by elastic links. Each point has mass m, and the spring constant of the links equals K. The location of the points at equilibrium, all lying in the same vertical plane P, can be found using the principle of minimum energy. Let $x^i \in \Re^2$ denote the coordinates of the ith point in P, $i = 1, ..., n$, with $x^0 = a$, $x^{n+1} = b$. The potential energy of the link between points i and $i+1$ equals $\frac{K}{2}\|x^{i+1} - x^i\|^2$. The potential energy of the ith mass point equals $mge^T x^i$, where $e^T = (0, 1)$ and g is the gravitation constant. The total energy is then

$$E(x^1, \ldots, x^n) = \frac{K}{2}\sum_{i=0}^{n} \|x^{i+1} - x^i\|^2 + mg\sum_{i=1}^{n} e^T x^i.$$

(a) Is E convex? Give a rigorous argument for your answer.

(b) Equilibrium is achieved at a global minimum of E. Show that, at equilibrium, the first coordinate of the differences $\Delta^i = x^{i+1} - x^i$ are constant, so the horizontal positions of the mass points are equally spaced. Show also that the second differences $\Delta^i - \Delta^{i-1}$ are constant, so the mass points are located on a parabola.