Math 270B: Numerical Approximation and Nonlinear Equations

Instructor: Michael Holst

Winter Quarter 2013

Homework Assignment #2
Due : Give to the class TA within a couple of weeks if you would like it marked before the final.

This homework covers the main material in 270B related to **nonlinear equations**. *Our goal is to study three related problems in nonlinear equations: solving systems of nonlinear equations, unconstrained optimization involving nonlinear functionals, and equality-constraint optimization involving nonlinear functionals and equality constraints. We will use many of the basic ideas about vector spaces and linear operators that we studied in the first homework, together with our background from 270A.*

Exercise 2.1. Let $F(x)$ denote a twice-differentiable function of one variable. Assuming only the mean-value theorem of integral calculus: $F(b) = F(a) + \int_a^b F'(t)\, dt$, derive the following variants of the Taylor-series expansion with integral remainder:

(a) $F(x+h) = F(x) + \int_x^{x+h} F'(t)\, dt$.

(b) $F(x+h) = F(x) + h\int_0^1 F'(x+\xi h)\, d\xi$.

(c) $F(x+h) = F(x) + hF'(x) + h\int_0^1 [F'(x+\xi h) - F'(x)]\, d\xi$.

(d) $F(x+h) = F(x) + hF'(x) + h^2\int_0^1 F''(x+\xi h)(1-\xi)\, d\xi$. *Hint: Try expanding $F'(x+h)$ using a formula like part (b) and then differentiate with respect to h using the chain rule.*

Exercise 2.2. Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a differentiable function. Using the multi-dimensional verson of the fundamental theorem of calculus, show (i.e. derive) that the following Taylor expansion holds with the given integral remainder:

$$F(x+h) = F(x) + F'(x)h + \int_0^1 \{F'(x+\xi h) - F'(x)\}\, h\ d\xi.$$

Note that this is the multi-dimensional analogue of what you showed in Exercise 3(c). While this particular form of the remainder will be the most useful to us in the case of $\mathbb{R}^n$, derive the multi-dimensional analogues of the other three forms of the remainder you derived in Exercise 2.1.

Exercise 2.3. Find the gradient vector $F(x) = \nabla f(x)$ of the following functions, and then find the Jacobian matrix of $F(x)$. (The Jacobian matrix of $F(x) = \nabla f(x)$ is the same as the Hessian matrix $\nabla^2 f(x)$ of $f(x)$).

(a) $f(x) = 2\big(x_2 - x_1^2\big)^2 + \big(x_1 - 3\big)^2$.

(b) $f(x) = (2x_1 + x_2)^2 + 4(x_2 - x_3)^4$.

Exercise 2.4. Find $f'(x)$, $\nabla f(x)$ and $\nabla^2 f(x)$ for the following functions of n variables.

(a) $f(x) = \frac{1}{2}x^T H x$, where H is an $n \times n$ constant matrix.

(b) $f(x) = b^T A x - \frac{1}{2}x^T A^T A x$, where A is an $m \times n$ constant matrix and b is a constant m-vector.

(c) $f(x) = \|x\| = \left(\sum_{i=1}^n x_i^2\right)^{1/2}$.

Hint: As in the examples I did in class, some of these derivatives are easier to compute using the Gateaux or variational derivative calculation, rather than keeping track of all of the indices when computing partial derivatives explicitly.

Exercise 2.5. Create a MATLAB m-file of the form:

```
function [F,J] = D(x)
  F = [ a ; b ];
  J = [ c d ; e f ];
```

where the expressions for a, b, c, d, e, f are chosen so that the function returns the 2×1-vector-valued function $F(x)$ and the 2×2 Jacobian matrix $J(x)$ for the function $F(x)$ from part (a) of Problem 2.3. Use this m-file to compute F and J at $x = (1, 0)^T$; and $x = (1, 1)^T$. Capture the output from the computation and turn it in with the homework.

Exercise 2.6. Give (precisely) the following definitions:

1. A direction of decrease for a function $f(x) : \mathbb{R}^n \mapsto \mathbb{R}$.
2. A descent direction for a function $f(x) : \mathbb{R}^n \mapsto \mathbb{R}$.
3. Lipschitz continuity of a function $F(x) : \mathbb{R}^n \mapsto \mathbb{R}^m$.

Exercise 2.7. Prove that a descent direction is always a direction of decrease. *Hint: I did this in class; it was based on Taylor expansion.*

Exercise 2.8. Calculate the following derivatives (show your work):

1. The derivative $f'(x)$, the gradient $\nabla f(x)$, and the Hessian matrix $\nabla^2 f(x)$ of the following real-valued function of three real variables:
$$f(x) = x_1^3 - 2x_2^2 + x_3$$
2. The jacobian matrix $F'(x)$ of the following function which maps $\mathbb{R}^n$ into $\mathbb{R}^m$:
$$F(x) = Ax - b, \qquad \text{where:} \quad A \in \mathbb{R}^{m \times n}, \ x \in \mathbb{R}^n, \ b \in \mathbb{R}^m.$$

Exercise 2.9. Using a one-dimensional Taylor series, derive Newton's method for solving $f(x) = 0$, where $f : \mathbb{R} \mapsto \mathbb{R}$. Draw a detailed picture for the case of $f(x) = x^2 - 9$, illustrating how the iterates behave, beginning with $x_0 = 10$. (The picture doesn't have to be pretty, but it must be logically correct and labeled correctly.) *Hint: I did this in class several times.*

Exercise 2.10. Let $F(x) : D \subset \mathbb{R}^n \mapsto \mathbb{R}^n$, and assume that the jacobian $F'(x) : D \subset \mathbb{R}^n \mapsto \mathbb{R}^{n \times n}$ is Lipschitz continuous with Lipschitz constant γ. Show that the error in the linear model
$$L_k(x) = F(x^k) + F'(x^k)(x - x^k)$$
of $F(x)$ can be bounded as follows:
$$\|F(x) - L_k(x)\| \leq \frac{1}{2}\gamma \|x - x^k\|^2.$$
Hint: I did this in class; just use the (multi-dimensional) Taylor formula you derived in Problems 2.1 and 2.2.

Exercise 2.11. Write a MATLAB m-file implementing Newton's method, using the MATLAB routines you wrote for problem 7. Apply Newton's method to part (a) of Problem 2.3 and collect the iteration information, using first the initial guess of $x = (1, 0)^T$, and then using the initial guess of $x = (1, 1)^T$.

Exercise 2.12. State and prove a basic convergence theorem for Newton's method for solving $F(x) = 0$, where $F : \mathbb{R}^n \to \mathbb{R}^n$. (See the notes from class regarding the minimal assumptions you will need to use.) First show that Newton's method converges superlinearly without assuming that $F'(x)$ is Lipschitz, and then use the result in Exercise 12 to show that Newton's method converges quadratically when $F'(x)$ is Lipschitz. *Hint: This is the main Newton's method theorem I stated and proved in class. If you know this theorem and how the proof works, then you can easily reconstruct the same argument for convergence of Newton's method for unconstrained optimization.*

Exercise 2.13. Given each of the following cases of a gradient $g(\bar{x})$ and Hessian $H(\bar{x})$ defined at a point $\bar{x}$, discuss the optimality of $\bar{x}$. I.e., check the first and second order conditions for optimality. (Do not use MATLAB, atleast for the 2x2 cases; you may need to know how to compute eigenvalues by hand for 2x2 cases in the final exam.)

(i) $$g(\bar{x}) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad H(\bar{x}) = \begin{pmatrix} 3 & 1 \\ 1 & 1 \end{pmatrix}.$$

(ii) $$g(\bar{x}) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad H(\bar{x}) = \begin{pmatrix} 3 & 2 \\ 2 & 0 \end{pmatrix}.$$

(iii) $$g(\bar{x}) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad H(\bar{x}) = \begin{pmatrix} 4 & -1 \\ -1 & 4 \end{pmatrix}.$$

(iv) $$g(\bar{x}) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad H(\bar{x}) = \begin{pmatrix} -2 & 0 \\ 0 & -3 \end{pmatrix}.$$

(v) $$g(\bar{x}) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \quad H(\bar{x}) = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

Exercise 2.14. Write a MATLAB function with specification `[f,g,H] = ex33(x)` that computes $f(x)$, $g(x)$ and $H(x)$ for the function

$$f(x) = e^{x_3}x_1^2 + 2x_2^2 + x_3^2 \cos x_1$$

at any point x. Use your function to compute $f(x)$, $g(x)$, and $H(x)$ at $x = (0,\ 0,\ 0)^T$ and $x = (-1,\ 2,\ -2)^T$. In each case, compute the spectral decomposition of the Hessian matrix and indicate if the necessary and sufficient conditions for unconstrained local minimization are satisfied.

Exercise 2.15. Let $q(x)$, $x \in \mathbb{R}^n$, be the quadratic function $q(x) = c^Tx + \frac{1}{2}x^THx$, where H is symmetric.

(a) Write down an expression for $\nabla q(x)$ in terms of c, H and x.

(b) Given an arbitrary point x_0 and a direction p, write down the Taylor-series expansion of $q(x_0 + p)$.

(c) For this part, consider $q(x)$ such that H is positive definite. If p is a direction such that $\nabla q(x_0)^T p < 0$, show that there exists a *positive* minimizer α^* of $q(x_0 + \alpha p)$. Derive a closed-form expression for α^*.

Exercise 2.16. Write a MATLAB m-file `steepest.m` that implements the method of steepest descent with a backtracking line search. Your function must include the following features.

- Use $\mu = \frac{1}{4}$ to define the sufficient-decrease criterion in the backtracking algorithm.
- The minimization must be terminated when either $\|g(x_k)\| \le 10^{-5}$ or 75 iterations are performed. Any MATLAB "`while`" loop must include a test that will terminate the loop if it is executed more than 20 times.

Use `steepest.m` to find a minimizer of the function

$$f(x) = e^{x_3}x_1^2 + 2x_2^2 + x_3^2 \cos x_1,$$

starting at the point $(-1,\ 1,\ 1)^T$ (first write a MATLAB function as in Exercise 2.3). Next, minimize the function (again, first write a MATLAB function as in Exercise 2.3)

$$f(x) = x_1 + x_2 + x_3 + x_4 + x_1^2 + x_2^2 + 10^{-1}x_3^2 + 10^{-3}x_4^2,$$

starting at the point $(-1,\ 0,\ 1,\ 1)^T$. Compare the two runs. Can you explain why steepest descent behaves like this?

Exercise 2.17. Modify the MATLAB m-file `steepest.m` from Exercise 2.16 to produce a MATLAB m-file `newton.m` that implements the Newton's method for optimization with a backtracking line search. Repeat the two examples in Exercise 2.5 using the `newton.m` m-file.

Exercise 2.18. Consider the nonlinearly constrained problem

$$\begin{array}{ll} \underset{x\in\mathbb{R}^2}{\text{minimize}} & 3x_2 + x_1^2 + x_2^2 \\ \text{subject to} & x_1^2 + (x_2+1)^2 - 1 = 0. \end{array} \tag{2.1}$$

(a) Show that $x(\alpha) = (\sin\alpha,\ \cos\alpha - 1)^T$ is a feasible path for the nonlinear constraint $x_1^2 + (x_2+1)^2 - 1 = 0$ of problem (2.1). Compute the tangent to the feasible path at $\bar{x} = (0,\ 0)^T$.

(b) If $f(x)$ denotes the objective function of problem (2.1), find an expression for $f\left(x(\alpha)\right)$ and compute $f\left(x(0)\right)$.

(c) Define the *Lagrangian function* $L(x, \lambda)$ and *constraint Jacobian* $J(x)$ for problem (2.1). Derive $\nabla L(x, \lambda)$, the gradient of the Lagrangian, and $\nabla^2_{xx}L(x, \lambda)$, the Hessian of the Lagrangian with respect to x.

(d) Determine whether or not the point $\bar{x} = (0,\ 0)^T$ is a constrained minimizer of problem (2.1).

Exercise 2.19. Consider the problem

$$\begin{array}{ll} \underset{x\in\mathbb{R}^2}{\text{minimize}} & x_1^2 + 2x_2^2 \\ \text{subject to} & x_1 + x_2 - 1 = 0. \end{array}$$

(a) Find a point satisfying the KKT conditions. Verify that it is indeed an optimal point.

(b) Repeat Part (a) with the objective replaced by $x_1^3 + x_2^3$.

Exercise 2.20. Write a MATLAB function that will compute $c(x)$ and $J(x)$ for the constraint function

$$c(x) = x_1 + x_2 - x_1x_2 - \frac{3}{2}.$$

Use your function to find $c(x)$ and $J(x)$ at $x = (.1,\ -.5)^T$, $x = (.5,\ -1)^T$ and $x = (1.18249728,\ -1.73976692)^T$. At each of these points, discuss the optimality of the constrained minimization problem:

$$\begin{array}{ll} \underset{x\in\mathbb{R}^2}{\text{minimize}} & e^{x_1}(4x_1^2 + 2x_2^2 + 4x_1x_2 + 2x_2 + 1) \\ \text{subject to} & x_1 + x_2 - x_1x_2 - \frac{3}{2} = 0. \end{array}$$