

AMSC 607 / CMSC 764 Homework 2, Fall 2008
Due 2pm Tuesday October 7

Assignment: Write a MATLAB function that minimizes a function using Newton's method with trust region.

Follow the directions in parts (ii), (iii), (iv), and (v) of Nash and Sofer's Problem 10.17, p.340 (the 106th page of the pdf file). (In my hard-copy of the book, it is Problem 16, p.327.) It begins, "Write a computer program for minimizing a multivariate function..."

Notes:

- Compute an eigendecomposition of each Hessian matrix and then use the method of Homework 1 to find the correct value of γ .
- Your implementation of Newton's method should be in a function `newtonmin` which can be easily applied to other functions $f(\mathbf{x})$ with $x \in \mathcal{R}^n$ and n arbitrary.
- The function `newtonmin` should be well-documented, as if it were a library code. In particular, **documentation at the top of the function** should include:
 - **purpose of code**, since this is certainly the first thing a user wants to know!
 - **name of author**, since it provides someone to whom bugs can be reported and questions asked.
 - **date of the original code and a list of later modifications**, since it gives information such as whether the code is likely to run under the current computer environment and whether it might include the latest advances.
 - **description of each input parameter**, so that a user knows what information needs to be provided and in what format.
 - **description of each output parameter**, so that a user knows what information will be yielded.
 - **brief description of the method and references**, to help a user decide whether the method fits his/her needs.

In-line documentation should identify the major sections of the code and provide some detail on the method used. It is important in specifying the algorithm, identifying bugs, and providing information to someone who might need to modify the software in order to solve a slightly different problem.

Note that the documentation should be an integral part of the code; in other words, it is not enough to include it in a separate document, because a potential user might not have access to that document.

- **Parameters for newtonmin:** One way to do it:
 - the name of a function that evaluates the function to be minimized
 - the name of a function that evaluates the gradient of the function
 - the name of a function that evaluates the Hessian matrix
 - an initial guess at the solution.
 - a convergence tolerance
 - a maximum number of iterations.

If you want to use a different set of parameters, that is fine, as long as `newtonmin` remains “general purpose”.

Note that in Matlab, you pass a function name `myfunction` using the parameter `@myfunction`, and you call the function using MATLAB’s `fval`.

- **Obtaining gradient and Hessian values:** Either program them yourself or find an automatic differentiation program (search “automatic differentiation matlab” on the internet). Two notes: automatic differentiation packages are a bit difficult to work with, and note that this is not the same as symbolic computation.

Maximum points:

- `newtonmin` correctness: 15 points
- `newtonmin` documentation: 10 points
- Correct implementation of the 5 test problems: 5 points
- Part (v): 20 points. Give special attention to this part. In particular,
 - For each function except the first, plot the iterates on a contour plot of the function (`contour`).
 - Comment on when the trust region prevents Newton from taking a full step.
 - Compute the convergence rate for each of the 5 test problems.
 - Comment carefully on whether each result is consistent with the convergence theory.

Submission: by hard-copy, given to me after class or delivered to my office. (I will ask for an electronic copy of your programs if I need them.) **Please staple in the following order:**

- Listings of each MATLAB function and script that you wrote.
- A MATLAB “diary” of the output.
- A copy of each plot.
- Your written answer, separately for each of the 5 functions, of how the results are consistent with the theory of Newton’s method.