Examples and practice: Gradient Descent algorithm for function minimization, optimization

C. Alex Simpkins

November 14, 2006

1 Introduction and motivation

There are many situations where we need to minimize a scalar function (a cost function typically) of one or more variables. Three motivating cases are:

1. Solving large linear systems of equations
2. Solving nonlinear systems of equations
3. Dynamical systems optimization and control

Let's develop the code, then do some simple examples.

2 Writing the gradient descent function/code

For a complete description of gradient descent, please refer to the numerical methods pdf on the web site, chapter 5. Here is the first code listing to start with. Type it in rather than copy-paste.

See the text documents accompanying this zip file.

3 Exercises/Practice

3.1 Exercise - solving linear systems of equations, n equations, n unknowns, positive definite A

Compute the gradient descent-based solution of the 2 simultaneous equations

$$4 * x1 + 1 * x2 = 3$$

$$1 * x1 + 5 * x2 = 3$$

Compare the solution (x(1) and x(2)) to the solution given by $A \backslash b$, using the same method of setting up A and b (in fact you can use the same variables and the one line command to compute $a = A \backslash b$, then compare a and x).

3.2 Exercise - Repeat above with another positive definite matrix A and another b

Create another matrix, which is symmetric, and use the following test for positive definiteness:

$$det(A) > 0?$$

- The matrix is positive definite if and only if the determinants of all N submatrices are positive.
- Also the element with the max magnitude must be along the diagonal
- The diagonal elements of a positive definite matrix are all positive

3.3 Exercise - Overdetermined systems

solving m equations, n unknowns, m¿¿n (such as having x and y data to fit similar to least squares)

When we have an overdetermined system (more data than parameters) we precondition the system by multiplying by the transpose of A both sides:

$$Ax = b$$

$$A^T * Ax = A^T * b$$

Then the coefficient matrix $A^T A$ becomes square, symmetric, and positive definite. This is called the 'normal' system. It satisfies the original system in the least squares sense.

use the data t=0:10; y=5*t+3;

Now to make this work with gradient descent we have to simply multiply both A and b by A' on the left:

$$A = A' * A$$

$$b = A' * b$$

3.4 Excercise - repeat above with new data you make up

Do the same as above with another set of data. Plot the results in terms of the residuals (res_save), the path of the coefficients, etc.