

CS726, Fall 2008
Homework 6
(due Friday 10/17/07)

Submit your Matlab files electronically using the instructions on the course web page. Once you have set up the path in your user directory, you need to put your files for submission in a single directory and run the command

```
handin -c cs726-1 -a hwk6 -d <directory name>
```

You should hand in exactly five files with the names `CG.m`, `CG_FR.m`, `CG_PRplus.m`, `SteepDescent.m`, and `comments6.txt`. The last file should contain your written responses to questions 1 and 2.

Your codes must be submitted electronically by 5:00pm on the due date. Written answers to questions 3 and 4 are to be submitted in class on the due date.

1. Program the conjugate gradient algorithm - Algorithm 5.2 in the text. Name your program `CG.m`, and test it using the file `hwk6_cg.m` on the class web site. The file `hwk6_cg.m` also indicates the argument list and return values you should use for your routine. Note that the first argument of `CG.m` should be a pointer to a routine that accepts as input a vector x and returns the product Ax . Declare convergence when $\|r_k\| \leq \text{CGparams.toler} * \|b\|$. Be careful not to perform any unnecessary matrix-vector multiplications or unnecessary storage of vectors.

Use the routine `Aexplicit.m` on the web site to provide the matrix-vector product Ax for any given x .

Comment on the convergence behavior of the method on each of the three test problems programmed in `hwk6_cg.m`.

2. Write efficient codes for the Fletcher-Reeves nonlinear conjugate gradient algorithm, the Polak-Ribière method with the modification (5.45), and the steepest descent method. For all methods, use the `StepSize.m` routine that you coded in earlier assignments to find an approximate α_k that satisfies the strong Wolfe conditions. Terminate when the following condition is satisfied:

$$\|\nabla f(x_k)\|_\infty \leq \text{nonCGparams.toler} * (1 + |f(x_k)|),$$

or else when the number of iterations exceeds `nonCGparams.maxit`, whichever comes first.

In your submitted codes, use the following parameter settings for `StepSize`:

```
{\tt params = struct('c1',0.01, 'c2',0.3, 'maxit',100)};
```

Test your codes using `hwk6_nonlinearcg.m`. Note that the function to be minimized is Powell's singular function, coded as `xpowsing.m`.

Try modifying the line search parameters to see if you can improve the performance of the algorithms, and comment on your experiences in the file `comments6.txt`.

3. Problem 5.2 from the text.

4. Problem 5.4 from the text.