MCEN 5228, Spring 2014

Homework 4 due in class on 03/07

In your write-ups, please provide clear explanations of the models chosen, of the equations used, and of the findings, with figures where necessary.

1. This problem is a variation on problem 6.5 from Aster. Data values of $\mathbf{G}$ and $\mathbf{d}$ are contained within the `blur.mat` file provided with the text.

   (a) Use the `spy` command to investigate and plot the $\mathbf{G}$ matrix. Explain the patterns you see.

   (b) MATLAB stores $\mathbf{G}$ as a sparse object, retaining only the non-zero values. You can determine the number of non-zero values using `nnz`. How many are there? Given that they are stored in single precision (32-bit), and recalling that there are 8 bits per byte, report the size of $\mathbf{G}$ in sparse format. How much memory would be required to hold the full $\mathbf{G}$ in memory? To hold the results of an SVD decomposition of $\mathbf{G}$ in memory? Note: if you would like to crash your computer, try `svd(full(G))`.

   (c) The following commands will be useful for plotting images. Use them to plot the raw, blurred image $\mathbf{d}$.

   ```
   img=reshape(d,100,100);
   figure(1)
   imagesc(img);
   colormap(gray);
   ```

   (d) Using the `cgls` program provided in the text, compute the CGLS solution for 100 iterations. Make plots of the residual error $||\mathbf{Gm} - \mathbf{d}||_2^2$ and the norm of the solution at each iteration.

   (e) Plot a few images at different iterations. Around what range of iteration values does your inverted solution look the best?

   (f) Modify the `cgls` program to solve the least squares problem using the Steepest Descent algorithm. Add the residual error and solution norms as a function of iteration to your figures from part (d).

   (g) MATLAB has a CG routine that uses preconditioning, `pcg`. Note that this routine is for CG, not CGLS, so you will need to transform your problem in order to apply this routine. Look at the help documentation for `pcg` and try using some of the preconditioning techniques described there for this image blurring prolem. How well do they work?