4. SVD for pattern recognition

We borrow the following exercise from a NGSSC (National Graduate School of Scientific Computation) course given by Lars Eldén. If you want to know more, read his recent paper: Lars Eldén: Numerical linear Algebra in data mining, Acta Numerica (2006), pp. 327-384. Algorithms of this kind are used to recognize handwritten digits in postal codes, zip codes in US Postal service terminology. The data is from their test of prospective algorithms.

Construct an algorithm in Matlab for character recognition of handwritten digits. Using a training set, compute an SVD of each matrix of digits of one kind. Use the first few (5-20) singular vectors as basis and classify unknown test digits according to how well they can be represented in terms of the respective bases (use the residual vector in the least squares problem as a measure). Try to tune the algorithm for accuracy of classification (varying the number of basis vectors). Check if all digits are equally easy or difficult to classify. Report the number of incorrectly classified digits in a table. Also look at some of those, and see that in many cases they are very badly written.

If time permits, check the singular values of the different digits, and see if it is motivated to use different numbers of basis vectors for different digits.

The test data can be fetched via the course homepage. Read the data with the command:

```matlab
load('zipdata.mat')
```

The following files are then loaded:

1. dtrain and atrain: the first is a vector that holds the digits (the number) and the second is an array of dimension $256 \times 1707$ that holds the training images. The images are vectors of dimension 256, that have been constructed from $16 \times 16$ images.
2. The test data are given in dtest and atest. Use your algorithm on the columns of the matrix atest to see if you guess the corresponding values in the vector dtest! There are 2007 digits in the test set.
3. There is a function ima2.m that takes an image vector as input and displays it.