

# Homework Zero, due Tue 1/17

CSE 250B

1. Acquire access to Matlab (or Octave, [www.octave.org](http://www.octave.org)) and start getting comfortable with it.

Before embarking upon the rest of this homework set, you will need to download the OCR data from the course webpage. There are 20,000 training samples and 5,000 test samples, divided over four files called `train-images`, `train-labels`, `test-images`, and `test-labels`. To load the training data into Matlab, use:

```
fp = fopen('train-images', 'r');
trainx = fread(fp, [20000 28*28], 'uchar');
fclose(fp);
fp = fopen('train-labels', 'r');
trainc = fread(fp, 20000, 'uchar');
fclose(fp);
```

This creates two arrays: a  $20000 \times 784$  array `trainx` containing the data points, one point per row, and a one-dimensional array `trainc` (of length 20000) containing the labels (0 – 9). The test data is similar, but contains only 5000 points.

In order to view one of the data points, you can use Matlab's `image` command, although you first need to `reshape` the format from a 784-dimensional vector into a  $28 \times 28$  matrix. For instance, to see point number 275, you would use:

```
image(reshape(trainx(275,:), [28 28]));
```

If you find the output somewhat garish, try prefacing the last command with:

```
colormap(1.0-gray);
```

2. *Gross statistics of the OCR data.* In this exercise, you will compute some very basic statistics of the OCR data, to get a better sense of its geometry. Let  $S_i$  denote the training samples for digit  $i$  ( $0 \leq i \leq 9$ ). We can think of each  $S_i$  as a cluster.

- (a) Within each cluster  $S_i$ , the *squared interpoint distances* are  $\|x - y\|^2$  for  $x, y \in S_i$ . There are  $\binom{|S_i|}{2}$  of them. Plot a histogram of these values (using Matlab's `hist` command) for each  $i$ .
- (b) For each cluster, compute the average squared interpoint distance. The higher this value, the greater the variance within the cluster. Notice that the disparity in these numbers indicates that different parts of the data-space are inherently at different *scales*.
- (c) A common model of a cluster is as a uniform distribution over the surface of a sphere. Which of these clusters best fit this model, and which ones deviate the most? Explain your reasoning.

3. *Nearest neighbors.*

- (a) Write a Matlab function for computing the  $k$  nearest neighbors of a query point  $z$ , along with their majority vote:

```
function [class, ids] = knn(X,C,z,k)
```

Here  $X$  is the training data (some  $n \times d$  array of  $n$  points in  $\mathbf{R}^d$ ) and  $C$  is a vector of training labels (of length  $n$ ). The values returned are `ids`, a vector of length  $k$  holding the indices (into  $X$ ) of the  $k$  nearest neighbors; and `class`, the majority vote over those neighbors.

- (b) One of the things your function does is to compute Euclidean distances from  $z$  to each row of  $X$ . Perhaps the most obvious way to do this is in a loop:

```
dist = zeros(1,n);
for i = 1:n
    dist(i) = norm(z, X(i,:));
end;
```

Here is a faster alternative. Start by precomputing an array `sqlength`:

```
sqlength = sum(X .* X, 2);
```

Now, when given a query point  $z$  (say in row vector format), use the following in place of the distance computation shown above:

```
dist = sqlength - 2 * X * z';
```

This does not actually compute Euclidean distance, but it does return the right answer when used in a nearest-neighbor procedure! Explain why. [*Hint*: use the expansion  $\|x - y\|^2 = \|x\|^2 + \|y\|^2 - 2x \cdot y$ .]

- (c) Plot the error rate of the nearest-neighbor classifier (on the test set) as a function of  $k$ , for  $k \in \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19\}$ . The Matlab `plot` function is useful for this. Which pair of digits seems hardest to distinguish?