IC250 Lab 3

Classification of sparse vectors

In many systems, the data to be processed is inherently sparse. Such sparse data is seen in applications such as time series analysis, modeling heat flows, communications networks, and signal processing.

In this assignment, you will read sparse vectors (denoted by $\mathbf{x}$) of dimension $N \times 1$. Most of the elements in these vectors will be zero. Each vector belongs to one of ten classes $C_1, C_2 \ldots C_{10}$. The vectors will have a small "band" of non-zero elements, which is characteristic of its class. For example, if $\mathbf{x} \in C_1$, then most (but not all) of the elements between indices $i = K$ to $i = K + W$ will be non-zero, for some K and W. The non-zero elements in any $\mathbf{x}$ are restricted to its corresponding band. This band information can be used to determine the class of each vector.

Problem: Given a reference vector $\mathbf{x}$, and a set of vectors $Y = \{\mathbf{y_1}, \mathbf{y_2} \ldots \mathbf{y_n}\}$, determine how many vectors in Y belong to the same class as $\mathbf{x}$. This can be done by determining if the vectors share the same non-zero band as $\mathbf{x}$.

Data: The data provided as part of this question contains two directories: `ref/` and `data/`, respectively for $\mathbf{x}$ and Y. For each vector in `data/`, check against the vector in `ref/`. Determine the total number of vectors in `data/` which match the vector in `ref/`. To test your program, several sets `ref/` and `data/` is provided. The provided data has 100-dimensional vectors.

Code: A basic sparse matrix ADT is given with this assignment (see the Moodle page.) This is mostly based on the code in Sahni's textbook. Alternate code, with slightly different functionality is available in the August 2015 IC250 Moodle page.

Note:

1. Since the vectors are sparse, a sparse representation must be used. Do not utilise arrays of size N.
2. The vectors contain floating-point numbers (not integers.)
3. Your program must be able to process any $\mathbf{x}$ and any Y (not just with the data given with the assignment.) You can assume that the dimensions will remain 100×1.

Optional

1. Implement the sparse matrix ADT yourself, do not use the provided library.
2. Implement the addition of sparse vectors. Use the data provided with the assignment.