CSC221 Data Structures & Algorithms I Fall 2016

Programming Assignment #2 – K-Means Clustering

In this lab our goals are to:

- further develop your C++ programming skills
- use doubly linked lists
- gain a gentle introduction to (unsupervised) machine learning

K-means clustering is a data processing technique for discovering groups within a (possibly large) data set. It is easiest to apply to a set of real-valued (floating point) multi-dimensional points, Variations on K-means clustering can also be used for ordered discrete (e.g., integer) data and for un-ordered data.

In this assignment we will perform K-means clustering on a data set containing measurements of three species of Iris: *Iris virginica*, *Iris setosa*, and *Iris versicolor*. While there are three species contained in the data set, two of the three species are not well distinguished by the measurement data. However, one species is well distinguished from the other two. We will use two clusters.

Data Set

The data consists of 150 lines contained in a text file. Each line has five numbers on it. From left to right these numbers are:

Observation number Sepal length Sepal width Petal length Petal width

The first seven lines of the input data file are shown below:

```
1    6.3    2.8    5.1    1.5
2    6.3    3.4    5.6    2.4
3    5.4    3.9    1.7    0.4
4    6.5    3.0    5.8    2.2
5    6.2    2.2    4.5    1.5
6    5.8    2.6    4.0    1.2
7    4.8    3.4    1.6    0.2
```

You can download the data set from:

http://menehune.opt.wfu.edu/csc221/Projects/Lab2/lab2_data.txt

Project Requirements:

1. Let us agree to name the completed executable **kmeans**.
2. Your main program accepts the name of the data file on the command line. E.g.,

```
kmeans  lab2_data.txt
```

3. Use doubly linked lists to represent the clusters.

4. Implement the k-means clustering algorithm.

5. Output two text files: `A.txt` and `B.txt` representing the result of k-means clustering. Each output file contains the observation numbers (printed one per line) of the data points of that cluster.

6. Use good organization for your source code. Implement the doubly linked lists as a class with a separate source and header files.

7. Your project must include a `Makefile` for compiling and linking your program .

Helpful Hints:

- Maintain two doubly linked lists to represent the clusters. For discussion purposes, let's call them `A` and `B`.
- The K-means clustering algorithm (with two clusters) is outlined below.

```
Create two lists:  A  and B  ;
Read the data file and assign each observation randomly
     to either A or B
Select two observations at random to serve as initial means ;
// Call them 'meanA' and 'meanB' ;
do {
     change = false ;

     Scan list A.  For each observation X on list A, compute the
     Euclidean distance from X to meanA and from X to meanB.
     If X is closer to meanB than meanA, then {
             Remove X from list A ;
             Add X to list B ;
             change = true   ;
     }

     Scan list B.  For each observation Y on list B, compute the
     Euclidean distance from Y to meanA and from Y to meanB.
     If Y is closer to meanA than meanB, then {
             Remove Y from list B ;
             Add Y to list A ;
             change = true   ;
     }

     recompute meanA and meanB ;

while ( change ) ;

Output list A to file 'A.txt' ;
Output list B to file 'B.txt' ;
```

Turn-In :

Keep all your work in a sub-directory named `Lab2`. Change to the parent directory of `Lab2` and create a tar archive of your work using the command:

```
% tar cf Lab2.tar Lab2
```

Upload the file `Lab2.tar` to your account on telesto.