CSE 5243
Justin Ziniel
March 6, 2017

Assignment 4

Due Date: 3/29/17

Description

This lab is divided into 2 parts. In the first part, you will implement the *k*-means clustering algorithm and test your program on three different datasets. In the second part, you will use any off-the-shelf clustering algorithm to cluster these same datasets.

- **Dataset 1**: `TwoDimEasy` - Two numeric independent variables; 2 true, relatively well-separated clusters; 300 examples.
- **Dataset 2**: `TwoDimHard` - Two numeric independent variables; 4 true, slightly overlapping clusters; 400 examples.
- **Dataset 3**: `Wine Quality` - Information about the Wine Quality dataset can be found at https://archive.ics.uci.edu/ml/datasets/Wine+Quality. Remove the `class` variable from the dataset, i.e., when doing clustering, do *not* use the `quality` attribute (the dependent variable). This attribute will be used for external validation of the clusters, as discussed below.

You are *highly* encouraged to work on this project as part of a team (max of 2 people per team).

Part 1 (75% of grade)

The program should accept as a parameter the number of clusters, *k*, specified by the user. With the *k*-means algorithm, implement the standard Euclidean distance measure.

The output of the program should consist of two columns:

1. The row ID
2. The cluster that each record belongs to, as determined by the clustering method.

In addition to turning in the program, you should create a report which includes the following:

A. Describe your program and how it works. Discuss the design decisions that you made, e.g., how did you handle empty clusters?

For Dataset 1 and Dataset 2:

B. Given that you know the true clusters, compute the *true* cluster sum of squared error (SSE), the *true* overall SSE and the *true* between-cluster sum of squares (SSB), for each dataset.

C. Run k-means 3 different times for each dataset (using different initial centroids, and assuming $k = 2$ and $k = 4$ for Datasets 1 and 2, respectively). For each run (i.e., estimate):

(i) Compute the SSE for each *estimated* cluster (and the *estimated* overall SSE) as well as the *estimated* between-cluster sum of squares (SSB).

(ii) Compute the average silhouette width for each cluster and the entire clustering.

(iii) Create a scatterplot, overlaying the *true* cluster with the *estimated* cluster produced by k-means.

(iv) Create a confusion matrix comparing the actual and predicted clusters.

(v) Discuss your observations on how these metrics vary as the initial centroids are changed.

D. For Dataset 1, change the number of clusters to $k = 3$. Likewise, for Dataset 2, change the number of clusters to $k = 3$. For these two different configurations (number of clusters), compute each cluster SSE, the overall SSE and the SSB, the scatterplot and confusion matrices (as in part C, parts 1 - 4). Analyze these results compared to part C above. Answer the question on whether changing the number of clusters changes the results, and if so, for better or worse?

For the Wine dataset:

E. Experiment with different numbers of clusters and compare your results. What conclusions can you draw from your analysis?

F. Provide detailed analysis of the results. What trends did you observe? Provide graphs and/or statistics to back up and support your observations. Do you have a preferred clustering?

G. Use the quality attribute for external validation. Compare your clustering results to this attribute.

Part 2 (25% of grade)

Using any off-the-shelf clustering method (e.g., in R, Matlab, Python, Weka, etc.), run cluster analysis on the three datasets; you do not need to write code to implement these additional clustering algorithms.

The report should include the following components:

A. Discuss the clustering method that you used and any parameter settings that you chose.

B. Present the results of your cluster analysis, using similar metrics that were already discussed in **Part 1**.

C. Compare the results of this clustering method to the *k*-means implementation from **Part 1**.

What to Submit

1. Code (from **Part 1**)
2. Makefile (if applicable)
3. Readme - contains all the important information about the directory, including how to run the program and how to view the resulting output.
4. Written Report (PDF format preferred)
    4.1. The report should be a maximum of 9 pages (14 pages for teams of 2).
    4.2. The report should be well-written. Please proof-read and remove spelling and grammatical errors and typos. *Writing and presentation will be part of your grade for this assignment.*

You do not need to turn in the output datasets, rather these will be obtained by running your code.

How to Submit

Please choose one of the following programming languages: C/C++, JAVA, Python, MATLAB, R. (If there is another language you would like to use, please check with me first). All the related files (except for the datasets) will be tarred in a `*.zip` file or `*.tgz` file, and submitted via Carmen. Please use this naming convention: "`Project4_Surnames_DotNumbers.zip`" or "`Project4_Surnames_DotNumbers.tgz`". The submitted file should be less than 5MB.

On Linux Systems (Mac OS, Ubuntu, RedHat, etc.)

[Source Code, BashScript (and Makefile, if needed), Readme.txt, Report] should be submitted. Do not submit the raw datasets.

The program should be able to run on a standard Linux system. Readme.txt tells me where to put input data (raw dataset) and where to find output data and how to interpret the output.

When I type the command "`bash BashScript`", the output would be generated.

On Windows systems

[Source Code, Readme.txt, Report] should be submitted. Do not submit the raw datasets.

The program should be able to run on a Win-7 system. Readme.txt tells me where to put input data (raw dataset) and where to find output data and how to interpret the output. Readme also tells me how to compile and run the program.

Miscellaneous

If you use C/C++, please make sure your program can get compiled and run on VS2010 or later version, or you can choose to use GNU C++ compiler.

If you use JAVA, please make sure your program can get compiled and run on Eclipse.