Data Mining (CSC 503/SENG 474)

Assignment 3 **Due on Wednesday, March 24th, 11:55pm**

Instructions:

- You must complete this assignment on your own; this includes any coding/implementing, running of experiments, generating plots, analyzing results, writing up results, and working out problems. Assignments are for developing skills to make you strong. If you do the assignments well, you'll almost surely do better on the final and beyond.
- On the other hand, you can certainly have high-level discussions with classmates and can post questions to the conneX forum. You can also discuss preprocessing (data normalization) with classmates. You also are welcome to come to office hours (or otherwise somehow ask me and the TAs things if you are stuck).
- You must type up your analysis and solutions; I strongly encourage you to use LaTeX to do this. LaTeX is great both for presenting figures and math.
- Please submit your solutions via conneX by the due date/time indicated above. This is a hard deadline. Any late assignment will result in a zero (I won't accept assignments even one day late). However, I will make an exception if I am notified prior to the deadline with an acceptable excuse and if you further can (soon thereafter) provide a signed note related to this excuse.

Introduction

This assignment has one part (there is no additional part for grad students, since I want you all to also focus on your projects). The theme of this assignment is clustering. You will be running two types of clustering methods, Lloyd's algorithm ("k-means") and hierarchical agglomerative clustering, on two datasets. Part of the purpose of the assignment is for you to implement k-means (you need to implement it, but you do not need to implement hierarchical agglomerative clustering). The other point of the assignment is to get experience running clustering methods and interpreting the results, including how to go from results to selecting a good number of clusters.

1 Algorithms, Data, and Analysis

In this assignment, you will be using the following two algorithms:

- **Lloyd's algorithm (k-means)**. You *do* need to implement Lloyd's algorithm yourself. You should use the Euclidean distance. You also need to implement two forms of initialization:

  1. uniform random initialization — here, the centers are initialized to a set of k distinct examples from the dataset drawn uniformly at random (i.e. each example has the same probability of being a center).
  2. k-means++ initialization — This was described in lecture in detail and also is described in the lecture slides (see the slides for Lectures 15 and 16), in the Kevin Murphy book (see Section 11.4.2.7), and in various online resources (including Wikipedia). I would like to stress again that you do need to implement this yourself.

- **Hierarchical agglomerative clustering**. You should use Euclidean distance for the dissimilarity measure between two examples. For the dissimilarity measure between clusters, you should use both *single linkage* and *average linkage*. You do *not* need to implement hierarchical agglomerative clustering. You can use an existing implementation (it is part of scikit-learn, for example).

Therefore, in total, you will be running experiments with four clustering methods: two variants of Lloyd's algorithm and two variants of hierarchical agglomerative clustering.

What to test on

You'll be running the above clustering methods on two datasets that have been provided as part of this assignment:

- The first dataset, `dataset1.csv`, consists of 3500 two-dimensional examples. These examples were generated by a Gaussian mixture model.

- The second dataset, `dataset2.csv`, consists of 14,801 three-dimensional examples. If you know how to do a 3D scatterplot (or learn how to do one), you can see some interesting structure here (I encourage you to explore the structure to better understand the data).

Here are the experiments that you should run:

For each dataset:

- For each version of Lloyd's algorithm, try different values of k (starting from 2 and increasing) and, for each value of k, run the method multiple times and pick the best result. Make a plot of the cost[1] as k increases. Using the plot, decide how many clusters to use, and explain your choice.
- For each version of hierarchical agglomerative clustering, somehow decide what final clustering to pick (by making a cut in the dendrogram). Explain your reasoning for deciding where to make a cut in the dendrogram. My advice is to use the function `dendrogram` from the scipy package `scipy.cluster.hierarchy`. Since the number of examples is very large, you might find the option `truncate_mode = "lastp"` useful, in order to avoid showing too many lower levels of the dendrogram.

Discuss the results of all the methods (there are 4 methods in total). Use plots and dendrograms to support your discussion. Put your analysis in a file called `report.pdf`. By now, you have a lot of experience discussing results, so I am giving less details on how to do this.

Final advice on the report.

Please make your analysis concise (yet thorough). Don't ramble, and don't make stuff up. Act as if you are a respected scientist presenting some work to your colleagues (and you want them to continue being your colleagues after the presentation).

What to submit

In all, you should submit (as a zip file):

- the file `report.pdf` explained above;
- a file called `README.txt` which contains instructions for running your code or gives proper attribution to any code/datasets you got from other sources (like the Internet, for example). For parts of the code where you mostly used existing code/software but needed to modify a few files, then give attribution and mention the files you modified.
- a file called `code.zip`, containing your code. Please organize this file well (embrace the idea of directories). In case you mostly used some existing code/software but needed to modify a few files, then just provide those files here, and make sure your `README.txt` mentions those files in relation to the existing code/software.
- any additional files that you need.

2 How you'll be marked

- For each of CSC 503 and SENG 474, the total number of marks is 100, and you receive 25 marks for your code; most of these marks are for implementing the various versions of Lloyd's algorithm (with both of the types of initialization that were asked for).
- The remaining 75 marks are for your analysis in `report.pdf`.

[1] Recall from lecture that the cost is the sum of the squared errors from each point to its cluster center.