

The assignment is due IN CLASS.

1. Recall that the  $k$ -means algorithm can be generalized beyond the squared Euclidean distance to use any Bregman divergence. Code up the  $k$ -means algorithm when the divergence used is the KL-divergence. This clustering objective function can be expressed as

$$\min_{\pi_1, \dots, \pi_k} \sum_{c=1}^k \sum_{\mathbf{x}_i \in \pi_c} KL(\mathbf{x}_i, \mathbf{m}_c),$$

where  $\pi_i$  denotes the  $i$ -th cluster. Assume that the data vectors  $\mathbf{x}_i$  are non-negative with  $\ell_1$  norm equal to 1 (and hence the means  $\mathbf{m}_c$  also have  $\ell_1$  norm equal to one).

2. Consider the following “local search” procedure: Given a clustering  $\pi_1, \pi_2, \dots, \pi_k$ , consider moving a point from cluster  $\pi_i$  to  $\pi_j$ . Calculate the change in the above objective function for such a move, and show that this change can be computed in  $O(d)$  time, where  $d$  is the dimensionality of the input data. Write code for a function that calculates the best possible move given a clustering.
3. Extend the above function to find a sequence of moves that improves the objective function the most, as follows: find the best move, record the change in objective function, and “simulate” that move. Repeat for a second move, and so on until  $L$  moves have been simulated. Finally, perform the sequence of the first  $p$  moves ( $0 \leq p \leq L$ ), where  $p$  is selected such that the resulting objective function is the lowest over all  $p$ .
4. Create an algorithm that alternates between running (a) the  $k$ -means algorithm until convergence and, (b) the above local search moves. The algorithm stops when neither method significantly improves the objective function. Run your resulting clustering algorithm on the music data set and the journal/department data set. The music data set is available at `www.cs.utexas.edu/users/kulis/dm2007/musicdata.mat`, and contains reviews of 270 music pieces. Cluster labels are available for the documents based on composer (`complab`) as well as the form/genre (`formlab`). For the journal/department data, cluster the journals/conferences using the data set `www.cs.utexas.edu/users/kulis/dm2007/instbyconf.mat`. Let the maximum number of local search moves  $L$  be 20. Compare your clustering results to the “true” clusters (when available) by computing the confusion matrix, and compute the objective function value of the resulting clusters. How do the results compare with and without local search? Also comment on the results you obtain for the journal/conference clustering.
5. Download `gmeans` from the web: `www.cs.utexas.edu/users/yguan/datamining/gmeans.html`, and repeat the above experiment. For the music matrix, download the CCS format (a sparse matrix format) for the data at `www.cs.utexas.edu/users/kulis/dm2007/musicdataaccs.tar`, and for the journal/department data, download `www.cs.utexas.edu/users/kulis/dm2007/instbyconf.tar`. How do the `gmeans` results compare in terms of accuracy and speed to the results given by your program?