Lab 06

For this homework, provide a single rendered R Markdown file in *pdf* format on crowdmark for the problems (you may render the R Markdown file to *html*, and then convert the *html* file to *pdf* using the print function on your web browser). Indicate your student number on the markdown file before the first section header, and make a section for each lettered part of each problem (i.e., '`# Problem 1a)`', '`# Problem 1b)`', '`# Problem 2a)`' etc.). Provide in the markdown *the final version of all of the code you wrote for this homework*, and make sure long lines of code are wrapped in the rendered *pdf*. If your Markdown file involves examining a large dataset, do not print the entire dataset to the markdown file in the steps of your solution (instead, suppress the output, or only show a small section of the data as an example). If you do the bonus problem, provide it as a separate R Markdown file in *pdf* format on crowdmark.

Problem 1: k-means

a) Write a function called *my.dist2*. This function should take two data frames, the first with N rows and the second with K rows. Both of the data frames should have the same column names (i.e., they should both be D dimensional). This function should return a matrix with N rows and K columns such that the i, j-th element is the Euclidean distance between the i-th row of the first data frame and the j-th row of the second data frame. (So, this function should operate the same way as the function *dist2* from the package *flexclust*.)

(2 points)

b) Write your own implementation of the k-means without using any libraries and without using the R function *kmeans*, as a function called *my.kmeans*. It should work for datasets of arbitrary dimension D, and

should return the centroids and the cluster assignments of the last iteration of the algorithm. You may work from the code from the slides for Week 08 (but extend from 2 dimensions to arbitrary dimension, and use your own *my.dist2* function). For stopping condition, have your function take two parameters (in addition to a parameter for the data): 1) A maximum number of iterations such that if the number of iterations reaches this maximum number, the iterations stop, 2) A threshold such that if the all Euclidean distances between the centroids before an iteration and after an iteration is less than this threshold, the iterations stop. Set reasonable default values for these parameters. Provide your code, and write a one paragraph *help* for your function, indicating the names of the parameters, and the nature of the return value.

(4 points)

d) Create a simulated dataset by hand with $k+1$ clusters using the Calm Code page (the link is in the Week 08 slides). Set k to be the last digit of your student number. Run both *my.kmeans* and *kmeans* on the data. Make scatter plots with the results. In at most a few sentences: Are the results for the two methods the same? Why or why not?

(4 points)