# CS 434: Assignment 4

Due May 19st 11:59PM, 2017

General instructions.

1. The following languages are acceptable: Java, C/C++, Matlab, Python and R.
2. You can work in team of up to 3 people. Each team will only need to submit one copy of the source code and report.
3. You need to submit your source code (self contained, well documented and with clear instruction for how to run) and a report via TEACH. In your submission, please clearly indicate your team members' information.
4. Be sure to answer all the questions in your report. Your report should be typed, submitted in the pdf format. You will be graded based on both your code as well as the report. In particular, the clarity and quality of the report will be worth 10 % of the pts. So please write your report in clear and concise manner. Clearly label your figures, legends, and tables.

## 1 Data description

In this implementation assignment you will explore hierarchical and non-hierarchical clustering. The provided data set contains handwritten digits. Data file has 30000 rows, where every row is a 784 dimensional vector that represents a particular digit. You may use imshow(reshape(x,28,28)) to plot the image.

## 2 Non-hierarchical clustering - K-Means algorithm

1. (25 pts) Implement K-means algorithm. Run your K-means algorithm with $k = 2$. To verify that your algorithm actually converges, please plot the objective of the K-means algorithm (i.e., the SSE) as a function of the iterations. From one run to another run, this curve may look different. Just present the results of a typical run.
2. (25 pts) Now apply your K-means implementation to this data with different values of $k$ (consider values $2, 3, \cdots, 10$). For each value of $k$, please

run your algorithm 10 times, each time with a different random initialization, record the lowest SSE value achieved in these 10 repetitions for each value of $k$. Plot the recorded SSE values against the changing $k$ value. What do you think would be a proper $k$ value based on this curve? Please provide justification for your choice.

## 3 Hierarchical agglomerative clustering (HAC)

1. (25 pts) Implement HAC algorithm using single link to measure the distance between clusters. Report the dendrogram starting with 10 clusters. More precisely, run HAC algorithm and wait until all the data points will be grouped into 10 clusters. Start building your dendrogram with those 10 clusters until you get only one cluster. Clearly indicate the heights of the tree branches, i.e., the distances between merged clusters at that particular step. Note, that your program does not have to automatically generate the dendrogram. You just need to record which clusters get merged at what height. Then you will produce the dendrogram manually based on the recorded merging process. You can use PowerPoint or any other software of your choice to draw dendrogram. Looking at the dendrogram, can you determine the number of clusters? Explain your choice.

2. (25 pts) Implement HAC algorithm using complete link to measure the distance between clusters. Report the dendrogram starting with 10 clusters. More precisely, run HAC algorithm and wait until all the data points will be grouped into 10 clusters. Start building your dendrogram with those 10 clusters until you get only one cluster. Clearly indicate the heights of the tree branches, i.e., the distances between merged clusters at that particular step. Note, that your program does not have to automatically generate the dendrogram. You just need to record which clusters get merged at what height. Then you will produce the dendrogram manually based on the recorded merging process. You can use PowerPoint or any other software of your choice to draw dendrogram. Looking at the dendrogram, can you determine the number of clusters? Explain your choice.