CS 434: Assignment 4

Due May 27th 11:59PM, 2018

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 6000 rows, where every row is a 784 dimensional vector that represents a particular digit. You may use the matlab function imshow(reshape(x,28,28)') to plot the image.

2 Non-hierarchical clustering - K-Means algorithm

1. (25 pts) Implement the 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 Principal Component Analysis

1. (15 pts) Implement Principal Component Analysis for dimension reduction. Specifically, your program needs to compute the mean and covariance matrix of the data, and use a off-the-shelf numerical package of your choice to compute the top ten eigen-vectors with ten largest eigen-values of the Covariance matrix. Report the eigen values in decreasing order.

2. (15 pts) Plot the mean image, and each of the top ten eigen vectors using matlab function imshow(reshape(x, 28,28)') (or other comparable functions in other languages). To make the image for eigen vectors viewable, you should re-scale each eigenvector by its maximum value. Inspect the resulting images, what do you think they capture?

3. (10 pts) Use the top 10 eigenvectors to project each image to 10 dimensions. Identify for each dimension the image that has the largest value in that dimension and plot it using imshow. Compare the image with its corresponding eigen-vector imagem, what do you observe?