Midterm: Bernoulli, K-mean and EM (Bernoulli).

The purpose of this project was to make sense of the data by using unsupervised methods by evaluating the efficiency of Maximum Likelihood estimation, K-means, and Expectation-Maximation (EM) algorithms on mixture and multivariate Bernoulli distributions. The data set for this experiment was the MINST Data set, a compilation of handwritten digits in which each image is composed of a 28x28 matrix of pixels. Each algorithm is performed individually into three different sections.

Maximum Likelihood Estimation:

Each image represents a 28x28 matrix where each cell value describes the color density. Because each cell value describes the color intensity, which can lead to large values, they are normalized to fit in the range [0,1]. The desired function to maximize is a multidimensional Bernoulli random variable which is defined as $p(x|\theta_j) = \theta_j^{x_j}(1-\theta)^{1-x_j}$. Assuming images are independent, the Maximum Likelihood estimation is performed individually; hence, the mean is calculated for all image pixels. Any value less than 0.5 is changed to 0 and 1 otherwise. The calculation followed the formula $\theta_{MLE} = \frac{\Sigma_{i=1}^{N} x_{ij}}{N}$, the sum of each cell is divided by the total number of images, here, 20,000.

K-Means Implementation:

Each data is assigned to K clusters. Implementing this algorithm required initializing random mu's values and creating three functions named calcSqDistance, determineRnk, and recalcMus. The function calcSqDistance finds the distance of each data point to each cluster. The second function determines the closest cluster of each point and sets to true the closest one and false the others. Finally, the recalcMus calculates each mean based on the results of the second function. Those functions are performed in a for-loop until the means converge. The iteration number is set arbitrarily; here, it was set to 1000. Lastly, each cluster is plotted. The resulting plots showed more precise digits. Additionally, some of the same numbers appeared in different clusters. This algorithm was run for both 10 and 20 clusters.

Expectation-Maximation (EM) algorithm:

The EM algorithm also required initializing random values and implementing sub-functions. Since latent variables are not observable, the expectation with the observable is taken as the initial inferences of both theta and pi parameters. The pi parameter (1xK) is used as a

prior, to infer the probability of likelihood of each data. The theta parameter (KxD) represents the mean of each cluster. The gamma distribution is first performed using the randomly initialized prior and means by applying Bayes' theorem. The previously mentioned functions are named calcRespons and recalcParams. The first function uses a Bernoulli distribution and inferred priors to determine the responsibility of each cluster to a data point. Then, recalcParams recalculates each parameter. To prevent long-running times, vectorization was used to calculate theta's values in each iteration. This method saved the runtime from finding the clusters. Like the K-means algorithm, the EM algorithm is performed in a for-loop with 1000 iterations and 10 clusters.

Results:

It is clear from the resulting plots in each section that K-means and EM algorithms are more efficient than MLE. However, the plots from K-means and EM were similar in blurriness. A possible issue might be a lack of optimization in the EM algorithm. The average runtime of EM was 16 minutes. In this implementation, the number of iterations was set to 1000. Also, a loop was used for the calcRespons function. This leads to a nested loop indirectly between the function and the number of iterations. To reduce it, convergence can be reduced to be more lenient.

Section 1: Single Bernoulli

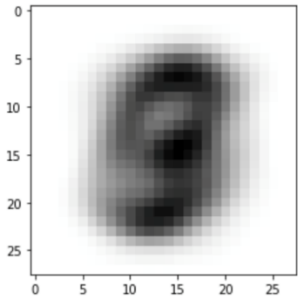

Figure 1: All images present.

The Single Bernoulli, this is the mean of all images present. It kind of resembles a 3.

A visualization of the θ_{MLE}.

Section 2: K-means

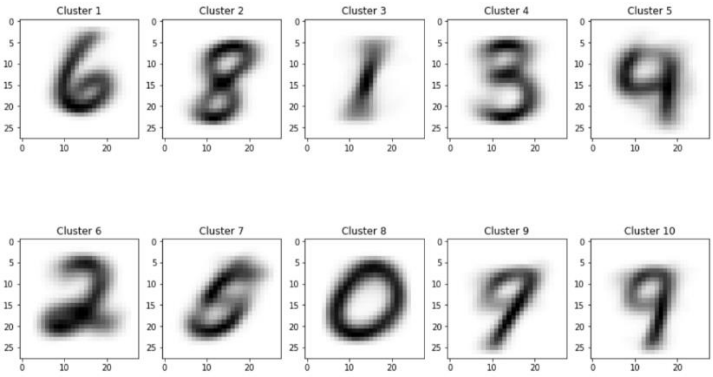

Figure 2: Plots for K=10 clusters

We can see there are 3 number 9's this can be due to the handwriting style.

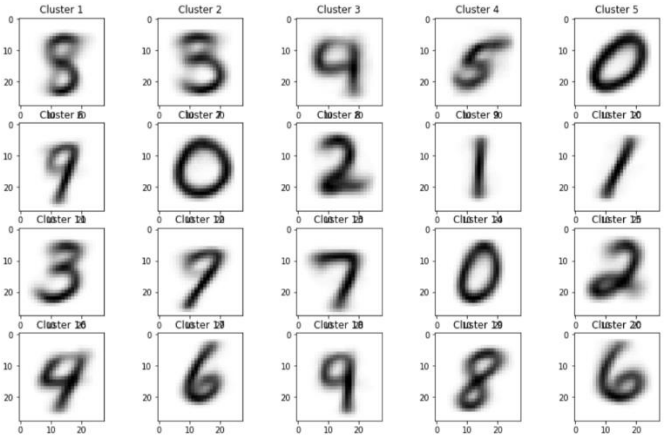

Figure 3: Plots for K=20 clusters.

K =20. Each cluster is different of the style each digit is written.

Section 3: EM

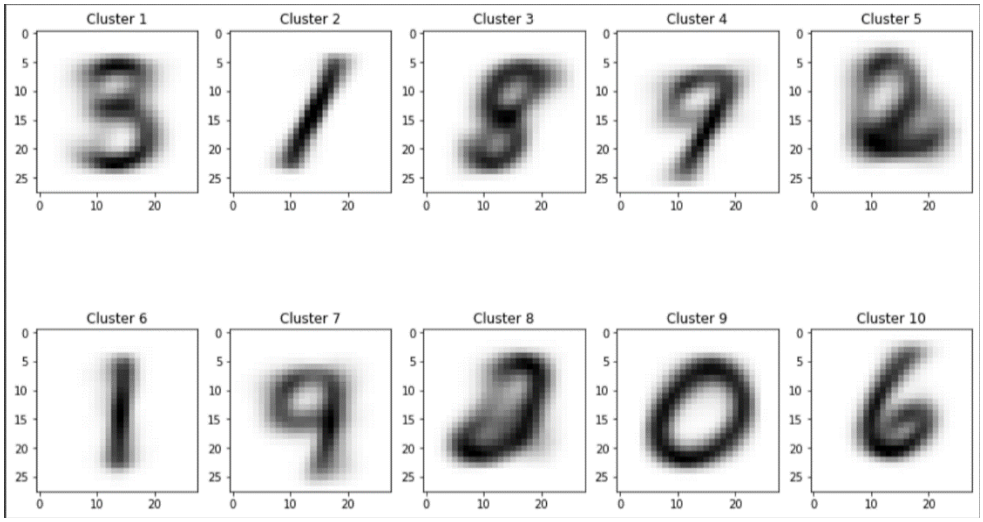

Figure 4: Plots for K=10 clusters using EM.

EM. The digits look slightly better, as there different clusters for each digit.