

IERG4300 / ESTR4300 Spring 2023

Homework 3

Release date: March 11, 2023

Due date: April 5, 2023 (Wednesday) 11:59 pm

No late homework will be accepted!

Every Student **MUST** include the following statement, together with his/her signature in the submitted homework.

I declare that the assignment submitted on the Elearning system is original except for source material explicitly acknowledged and that the same or related material has not been previously submitted for another course. I also acknowledge that I am aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained in the website

<http://www.cuhk.edu.hk/policy/academichonesty/>.

Signed (Student _____) Date: _____

Name _____ SID _____

Submission notice:

- Submit your homework via the elearning system

General homework policies:

A student may discuss the problems with others. However, the work a student turns in must be created COMPLETELY by oneself ALONE. A student may not share ANY written work or pictures, nor may one copy answers from any source other than one's own brain.

Each student **MUST LIST** on the homework paper the **name of every person he/she has discussed or worked with**. If the answer includes content from any other source, the student **MUST STATE THE SOURCE**. Failure to do so is cheating and will result in sanctions. Copying answers from someone else is cheating even if one lists their name(s) on the homework.

If there is information you need to solve a problem but the information is not stated in the problem, try to find the data somewhere. If you cannot find it, state what data you need, make a reasonable estimate of its value and justify any assumptions you make. You will be graded not only on whether your answer is correct but also on whether you have done an intelligent analysis.

Q1 [20 marks]: Parameter Design for Minhash/ Locality-Sensitive Hashing (LSH)

Let r be the number of rows within each band and B be the total number of bands within the Minhash signature matrix M (r and B are all integers). We want to design a system so that:

- 1) For any pair of items with a similarity greater than or equal to T_1 , the probability that they will be correctly identified as a similar-pair candidate should be at least P_1 .
- 2) For any pair of items with similarity below T_2 , the probability that they will be mistakenly identified as a similar-pair candidate should be no more than P_2 .

- (a) **[10 marks]** Derive the set of inequalities to govern the relationship between T_1 , T_2 , P_1 , P_2 , r and B so that the aforementioned accuracy/ error requirements would be satisfied.
- (b) **[10 marks]** Suppose each signature is represented by a 100-dimension vector. For $T_1=0.8$, $T_2=0.3$, $P_1=0.98$ and $P_2=0.02$, use your results in part (a) to derive a pair of values for (r, B) so that the aforementioned accuracy/ error requirements would be satisfied. If no (r, B) pair can satisfy the requirements, please explain your reasons. Show your steps (and source code if any).

Q2 [60 marks + 20 bonus]: K-means Clustering

The MNIST database is a dataset of handwritten digits, comprising 60000 training examples and 10000 test examples. Each instance corresponds to a handwritten digit. In this question, we will implement the K-means algorithm using MNIST dataset. The data can be downloaded from ref [1].

The MNIST dataset contains 10 various digits, with representative images shown in Fig. 1. Each of the digits is a 28x28 pixel image, resulting in a 784-dimensional space. In the provided MNIST dataset [1], there are four files:

- (1) train_img: training set images
- (2) test_img: test set images
- (3) train_label: training set labels
- (4) test_label: test set labels

File (1) and (2) contain image instances. Each row of the file corresponds to an image instance. For computation convenience, each image instance is reformatted as a 784-dimension vector (by concatenating the pixels in an image row-by-row). Each element/ pixel value (comma separated) ranges from 0 to 255. File (3) and (4) are the labels with respect to file (1) and (2). Each row contains a digit, which is the label corresponding to the same row in file (1) or (2).

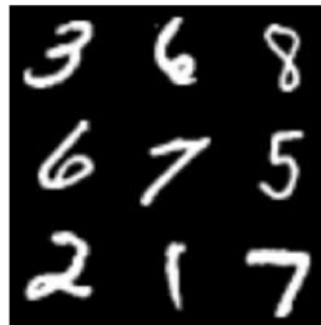


Fig. 1. An example of representative digits from the MNIST dataset.

(a) **[30 marks]** Referring to the lecture notes, you will need to implement K-means using MapReduce to perform clustering for the training set. For this question, the number of clusters is set to $K=10$. Please output the vector representation of each centroid and the number of image instances assigned to each cluster like the following example:

[Centroid ID]: [Number of Digit Images], [Centroid Vector]

Centroid 0: 1000, [784-dimension vector]

Centroid 1: 2000, [784-dimension vector]

.....

Centroid 9: 3000, [784-dimension variable]

Submit your steps, codes and results in a single PDF file.

Hint:

1. You should use multiple rounds of MapReduce to implement the K-means algorithm. For each round, each mapper processes part of the training data points and stores all the current centroids. The reducers will update the centroids based on the “partial sum” information collected from all mappers.
2. You don't need to remember (or store) the cluster-membership assignment of each training image instance. You can keep track of enough information to enable the computation of the centroid at the reducer (e.g., the partial sum of the training data points in a mapper for each cluster, and the number of training image instances in a mapper that is assigned to a specific cluster, etc.). You need to store/ output the cluster membership assignment for each data point after K-means converges. You are allowed to implement an extra program to assign each training image instance to the final clusters produced by K-means.

(b) **[20 marks]** In the simplest K-means algorithm, we initialize the centroids randomly. However, we know that a bad choice of centroids will lead to suboptimal clustering. One approach to avoid this situation is to test out different centroid initializations **and then choose an initialization that shows the best performance**. In this part, you will need to run K-means 4 times to find a good centroid initialization. For the first two runs, you should use (two different) randomly initialized centroids to start. Any reasonable random initialization scheme is acceptable. For the third and fourth runs, you will need to use the centroids (seeds) provided in [2] for initialization. You should read the *readme.txt* file in [2] for centroid format details.

By utilizing the clustering results of these 4 runs together with the ground truth labels, we can calculate the accuracy of the clustering results of the **training set**. The ground truth labels of training images are stored in the file (3) *train_label*, so you can compare the results with the labels. Following these steps and requirements to report your results:

1. Find the ground truth label of each image from file *train_label*.
2. Determine the label of a cluster by the majority of ground-truth labels in this cluster. For example, in a cluster, there are ten images with the label '4', five images with the label '3' and five images with the label '7', then the label of this cluster is '4'.
3. Calculate the accuracy of each run (ratio of correctly clustered images to total images): if the ground truth label of an image is the same as its cluster label, then it is correctly clustered. Otherwise, the image is clustered incorrectly.
4. Report the accuracy performance in the tables below with different "centroid initialization".
5. Compare the results in Table 1 to Table 4, and determine the best random seed. **Explain your choice.**
6. The submitted result should be in the same format as Table 1 to Table 4.

[Note: For part (b), you can implement the program either using a single machine or through MapReduce.]

Submit your codes, results and observations in a single pdf file.

Table. 1 The Accuracy of Clustering Performance with Random Seed 1

Cluster Index	Major Label of the cluster	# Train images belongs to the cluster	# Correctly clustered images	Classification Accuracy (%)
0				
1				
.....				
9				
Total Set	N/A			

Table. 2 The Accuracy of Clustering Performance with Random Seed 2

Cluster Index	Major Label of the cluster	# Train images belongs to the cluster	# Correctly clustered images	Classification Accuracy (%)
0				
1				
.....				

9				
Total Set	N/A			

Table. 3 The Accuracy of Clustering Performance with the Provided Seed 1

Cluster Index	Major Label of the cluster	# Train images belongs to the cluster	# Correctly clustered images	Classification Accuracy (%)
0				
1				
.....				
9				
Total Set	N/A			

Table. 4 The Accuracy of Clustering Performance with the Provided Seed 2

Cluster Index	Major Label of the cluster	# Train images belongs to the cluster	# Correctly clustered images	Classification Accuracy (%)
0				
1				
.....				
9				
Total Set	N/A			

(c) **[10 marks]** You should use the model with the best centroid initialization (the random seed that you have chosen from part (b)) throughout this question. In this question, you need to use the best clustering results to calculate the accuracy of the clustering results on the **test data**. Following these steps to report your results:

1. Find the test images from file *test_img*.
2. Assign each test image to the closest cluster, and assign a predicted label to the test image using the same label of the cluster.
3. Find the ground-truth label of each test image, calculate the classification accuracy by comparing the predicted label and the ground-truth label.
4. Report the accuracy of each cluster following the format of Table 5.

Table. 5 The Accuracy of Clustering Performance on the **Test Data**

Cluster Index	The label of the cluster	# test images in the cluster	# Correctly clustered images	Classification Accuracy (%)
0				
1				
.....				
9				
Total Set	N/A			

(d) **[bonus 20 marks]** In the above experiments with $K=10$, you may find the accuracy of a cluster is not as high as expected. In this part, you will need to evaluate the effect of K on accuracy by assigning different values of K for the K-means. Specifically, re-implement part (a) by using $K=12$, $K=15$ and $K=20$ respectively. For each different K , you should run at least three times, each of which should use different random centroid initializations. In each run, follow the steps in part (c) to record the classification accuracy of the **test set**. Please report your test accuracy results of the total set in each run following the format of Table 6. Compare them with the results in part (c) and explain your observations.

Table. 6 The Accuracy (%) on the **Test Set** with Different Settings of K

Accuracy (%) of K	$K=12$	$K=15$	$K=20$
1st run			
2nd run			
3rd run			
Averaged Accuracy			
Best Accuracy			

Q3 [20 marks]: Bernoulli Mixture Models

A Bernoulli Mixture Model (BMM) is a probabilistic model that assumes data

points are sampled from a mixture of multi-dimensional Bernoulli distributions. In what follows, we will derive the optimal parameters for BMM which would maximize its log-likelihood function:

Consider a set of binary variables x_i , where $i = 1, \dots, D$, each of which is governed by a Bernoulli distribution with parameter q_i , so that

$$p(\mathbf{x}|\mathbf{q}) = \prod_{i=1}^D q_i^{x_i} (1 - q_i)^{(1-x_i)}$$

where $\mathbf{x} = (x_1, x_2, \dots, x_D)^T$ and $\mathbf{q} = (q_1, \dots, q_D)^T$. In other words, \mathbf{x} follows a D-dimensional Bernoulli distribution where each variable x_i is independent of each other and $\text{Prob}(x_i=1) = q_i$ which is the i -th element of \mathbf{q} . Consider a mixture of K of such D-dimensional Bernoulli distributions with its density function given by:

$$p(\mathbf{x}|\mathbf{q}, \boldsymbol{\pi}) = \sum_{k=1}^K \pi_k p(\mathbf{x}|\mathbf{q}_k) \quad (*)$$

where $\mathbf{q} = \{\mathbf{q}_1, \dots, \mathbf{q}_K\}$ and $\boldsymbol{\pi} = (\pi_1, \pi_2, \dots, \pi_K)$. Now, consider a data set $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ which is generated by the Bernoulli Mixture model of (*).

Now, we show that the log-likelihood of $p(\mathbf{X})$ is given by:

$$\sum_{n=1}^N \log \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \mathbf{q}_k). \text{ First note that}$$

$$P(\mathbf{X}) = \prod_{n=1}^N \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \mathbf{q}_k)$$

By taking the logarithm of the above expression, we have:

$$\begin{aligned} \log P(\mathbf{X}) &= \log \prod_{n=1}^N \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \mathbf{q}_k) \\ &= \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \mathbf{q}_k) \end{aligned}$$

Define a variable $\gamma(z_{nk}) = \frac{\pi_k p(\mathbf{x}_n | \mathbf{q}_k)}{(\sum_{j=1}^K \pi_j p(\mathbf{x}_n | \mathbf{q}_j))}$, which represents the “responsibility”

of the k – th cluster (i.e. the k – th component of the Bernoulli mixture) for the data point (vector) \mathbf{x}_n . Now prove that the best π_k after the first round of

the EM-algorithm is $\frac{\sum_{n=1}^N \gamma(z_{nk})}{N}$ and $\mathbf{q}_k = \frac{\sum_{n=1}^N \gamma(z_{nk})x_n}{\sum_{n=1}^N \gamma(z_{nk})}$. In order to maximize with

respect to π_k , we need to introduce a Lagrange multiplier to ensure that

$\sum_k \pi_k = 1$. As a result, we now maximize the following quantity:

$$\log P(X) + \lambda \left(\sum_{k=1}^K \pi_k - 1 \right) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p(x_n | q_k) + \lambda \left(\sum_{k=1}^K \pi_k - 1 \right)$$

Taking derivative the above expression with respect to π_k , we have

$$\begin{aligned} \frac{\partial}{\partial \pi_k} \left(\log P(X) + \lambda \left(\sum_{k=1}^K \pi_k - 1 \right) \right) &= \frac{\partial}{\partial \pi_k} \left(\sum_{n=1}^N \log \sum_{k=1}^K \pi_k p(x_n | q_k) + \lambda \left(\sum_{k=1}^K \pi_k - 1 \right) \right) \\ &= \sum_{n=1}^N \frac{\partial}{\partial \pi_k} \log \sum_{k=1}^K \pi_k p(x_n | q_k) + \lambda \\ &= \sum_{n=1}^N \frac{p(x_n | q_k)}{\sum_{k=1}^K \pi_k p(x_n | q_k)} + \lambda \\ &= \sum_{n=1}^N \frac{\gamma(z_{nk})}{\pi_k} + \lambda \end{aligned}$$

Set it to 0, multiply both side by π_k and then sum over k , we have

$$\begin{aligned} \sum_{n=1}^N \frac{\gamma(z_{nk})}{\pi_k} + \lambda &= 0 \\ \sum_{n=1}^N \gamma(z_{nk}) + \lambda \pi_k &= 0 \\ \sum_{k=1}^K \left(\sum_{n=1}^N \gamma(z_{nk}) + \lambda \pi_k \right) &= 0 \\ N + \lambda &= 0 \\ \lambda &= -N \end{aligned}$$

Substituting $\lambda = -N$,

$$\sum_{n=1}^N \frac{\gamma(z_{nk})}{\pi_k} + \lambda = 0$$

$$\sum_{n=1}^N \frac{\gamma(z_{nk})}{\pi_k} - N = 0$$

$$\pi_k = \frac{\sum_{n=1}^N \gamma(z_{nk})}{N}$$

Finding q_k (Here we consider x_n and q_k as scalar, and the conclusion is also true is x_n and q_k are vectors)

$$\begin{aligned} \frac{\partial}{\partial q_k} p(x_n | q_k) &= x_n q_k^{x_n-1} (1-q_k)^{1-x_n} - (1-x_n) q_k^{x_n} (1-q_k)^{-x_n} \\ &= q_k^{x_n-1} (1-q_k)^{-x_n} (x_n (1-q_k) - (1-x_n) q_k) \\ &= q_k^{x_n-1} (1-q_k)^{-x_n} (x_n - q_k) \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial q_k} \log P(X) &= \frac{\partial}{\partial q_k} \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p(x_n | q_k) \\ &= \sum_{n=1}^N \frac{\pi_k}{\sum_{k=1}^K \pi_k p(x_n | q_k)} \cdot \frac{\partial}{\partial q_k} p(x_n | q_k) \\ &= \sum_{n=1}^N \frac{\pi_k}{\sum_{k=1}^K \pi_k p(x_n | q_k)} \left(q_k^{x_n-1} (1-q_k)^{-x_n} (x_n - q_k) \right) \\ &= \sum_{n=1}^N \gamma(z_{nk}) \cdot \frac{\left(q_k^{x_n-1} (1-q_k)^{-x_n} (x_n - q_k) \right)}{p(x_n | q_k)} \\ &= \sum_{n=1}^N \frac{\gamma(z_{nk}) (x_n - q_k)}{q_k (1-q_k)} \end{aligned}$$

Set $\frac{\partial}{\partial q_k} \log P(X) = 0$,

$$\begin{aligned} \sum_{n=1}^N \frac{\gamma(z_{nk}) (x_n - q_k)}{q_k (1-q_k)} &= 0 \\ \sum_{n=1}^N \gamma(z_{nk}) (x_n - q_k) &= 0 \\ \sum_{n=1}^N \gamma(z_{nk}) x_n - \sum_{n=1}^N \gamma(z_{nk}) q_k &= 0 \\ q_k &= \frac{\sum_{n=1}^N \gamma(z_{nk}) x_n}{\sum_{n=1}^N \gamma(z_{nk})} \end{aligned}$$

(a) **[20 marks]** Provide the pseudo-code (MapReduce is NOT required) to estimate the parameters of a BMM model based on maximum-likelihood arguments using the Expectation-Maximization algorithm. The pseudo-code should include a detailed description of the list of input/ intermediate variables used and how each of them gets updated during the E-step and M-step of each iteration.

References:

[1] The MNIST dataset:

http://mobitec.ie.cuhk.edu.hk/ierg4300Spring2023/static_files/homework/MNIST.rar

[2] Two seeds for K-means initialization:

http://mobitec.ie.cuhk.edu.hk/ierg4300Spring2023/static_files/homework/random_seeds.rar