DDA3020 Homework 4

Due date: Dec 13, 2023

Instructions

- The **deadline** is 23:59, Dec 13, 2023.
- The weight of this assignment in the final grade is 15%.
- **Electronic submission**: Turn in solutions electronically via Blackboard. Be sure to submit your homework as a single file. Please name your solution file as $A1\_studentID\_name$
- Note that **late submissions** will result in discounted scores: 0-24 hours $\rightarrow$ 80%, 24-120 hours $\rightarrow$ 50%, 120 or more hours $\rightarrow$ 0%.
- Answer the questions in English. Otherwise, you'll lose half of the points.
- Collaboration policy: You need to solve all questions independently and collaboration between students is **NOT** allowed.
- If you have any questions concerning this homework, feel free to reach out to TA. For Written parts, you may refer to TA Fanzeng Xia (223040232@link.cuhk.edu.cn). For Programming parts, you may refer to TA Xiaozhuang Song (xiaozhuangsong1@link.cuhk.edu.cn). You're also welcome to physically visit them during their office hours with your questions.

1 Written Problems (50 points)

1. (10 points) Show that the M step for maximum likelihood estimation of GMM(Gaussian Mixture Model)
2. (20 points) Consider a set of binary samples indexed by $i = 1, ..., m^+$ for positive class and $j = 1, ..., m^-$ for negative class.

    - Introduce the calculation principles of precision, recall, roc, and auc respectively.
    - When the distribution of positive and negative samples is uneven, which one is more affected, ROC or PR curve? Please explain mathematically.

3. (20 points) Consider the following 10 data points: $X = \{(1,0,2,-3,-2),(0,1,-3,-2,-3),(1,2,1,3,-2),(-1,1,2,3,-1),(1,0,1,-1,1),(2,3,-1,1,-2),(-2,3,-3,2,3),(-2,-2,2,3,-2),(-2,-2,1,-3,-3),(-3,2,0,-1,-2)\}$. Compute the unit-length principle components of X and choose two of them for PCA, then calculate the projection of each data on these two principal components. You could use python or matlab to obtain eigenvectors and eigenvalues. end

2 Programming (50 points + (10 points bonus))

Task Description:

In this programming task, you are required to

1. Implement 2 clustering algorithms **from scratch** (*i.e.*, no third-party or off-the-shelf package or library are allowed): (1) **K-means**; (2) **Gaussian Mixture Models (implement GMMs based on the EM algorithm discussed in class)**.

2. Implement 3 evaluation metrics **from scratch** (*i.e.*, not calling off-the-shelf package) to evaluate the performance of above clustering algorithms, including **Silhouette Coefficient**, **Rand Index** and **Normalized Mutual Information (NMI)**. Silhouette Coefficient and Rand Index have been introduced in the course, and you are required to implement a new metric Normalized Mutual Information, which will be introduced briefly in the below text.

3. Complete Task 2.1 and Task 2.2, and save and submit the assignments according to the requirements in the submission format.

4. (6 points bonus) For students who are willing to and successfully implement accelerated K-means with triangle-inequality (refer to `https://www.aaai.org/Papers/ICML/2003/ICML03-022.pdf`), they can receive a bonus of 8 points (Please place it in a separate file, and the total score not exceeding the overall score of this assignment).

5. (4 points bonus) We encourage conducting additional experimental analyses on the models, such as testing the sensitivity to clustering initialization for each algorithm (e.g., running a clustering algorithm with random initialization multiple times and calculating the variance of evaluation metrics). For students who provide comprehensive analyses, maintain a well-organized format and excellent presentation report, an additional bonus of 5 points can be awarded.

Task 2.1

Implement K-means and GMMs, as well as the 3 evaluation metrics: Silhouette Coefficient, Rand Index, and NMI.

Task 2.2

Plotting a figure that illustrate the values of the Silhouette Coefficient and Rand Index for different values of k.

Choose the optimal number of clusters k based on the Silhouette Coefficient and Rand Index (Please note that for this assignment, we do not consider the complex scenario of combining them to find an optimal k. In other words, we expect you to apply the Silhouette Coefficient and Rand Index separately to independently determine the optimal cluster number (k) for each metric.).

Then, calculate and output the NMI value for the selected k for each of the 2 metrics (Silhouette Coefficient and Rand Index).

Introduction to Normalized Mutual Information (NMI): Normalized Mutual Information (NMI) is a metric commonly used in information theory to quantify the similarity or dependence between two sets of data, often denoted as X and Y. It is particularly useful in clustering and classification tasks to assess the quality of partitioning or grouping of data points.

The formula for calculating NMI involves three key components: Mutual Information ($I(X;Y)$), Entropy of X ($H(X)$), and Entropy of Y ($H(Y)$). The NMI is computed using the following formula:

$$NMI(X,Y) = \frac{I(X;Y)}{\sqrt{H(X) \cdot H(Y)}},$$

where $I(X;Y)$ is the Mutual Information between X and Y

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) \cdot \log \left(\frac{P(x,y)}{P(x) \cdot P(y)} \right),$$

$H(X)$ and $H(Y)$ are the Entropy of X and Y respectively,

$$H(X) = -\sum_{x \in X} P(x) \cdot \log(P(x))$$

In this experiment, you are required to implement NMI and use the results obtained from your implemented clustering algorithm to calculate the NMI value for the results. Specifically, assuming you have the true class label set Y and the clustering algorithm-generated class label set X, the NMI formula allows you to assess the performance of the clustering algorithm on the data.

Datasets

We will be working with 2 datasets in this assignment:

1. **seeds:** 210 instances, 7 features, and 3 classes.
2. **Vowel:** 990 instances, 10 features, and 11 classes.

Submission Format:

1. For task 2.1, you are required to submit a code file (either a Python script or a Jupyter Notebook). **The code should include implementations of K-means and GMMs**, as well as the implementations of **Silhouette Coefficient, Rand Index, and NMI**. The execution of the code should output the values of the 3 mentioned evaluation metrics for k value range from 1 to 3 of the clustering number k on both 2 datasets.
2. For task 2.2, you need to submit a Jupyter Notebook file. The code should encompass the procedures outlined in task 2.2. The experiment is to be conducted solely on the Vowel dataset. You are free to define the range for selecting the k values, but please ensure that it includes at least 5 values for k.

3. For students seeking bonus points, you can keep the bonus-related scripts (accelerated K-means with triangle-inequality and additional experimental analyses) in a separate Jupyter Notebook or .py file.

Please name your code file as `A4_yourstudentID_2.1`, `A4_yourstudentID_2.2` (and `A4_yourstudentID_bonus` for those who complete bonus tasks), while `A4_yourstudentID` is your student ID number. For example, if your ID is `230056789` and submit 2 Jupyter Notebooks for task2.1 and task2.2, then you should submit `A4_230056789_2.1.ipynb` and `A4_230056789_2.2.ipynb`.

And again, please note that readable code is essential. If your code lacks comments and is intensively intricate, making it difficult to read, you may lose points.