CSE8803 - Homework 3 - Spring 2022

Please submit your answers to Questions 1 and the analysis part in Question 2 and 3 in one PDF file, and all the code that is required to submit in a separate .zip file.

1 Theory: Guassian Mixture Model [10 pts]

In the lecture, we discussed that the probabilistic version of the K-means algorithm corresponds to a simplified Gaussian Mixture Model (GMM). The standard GMM for clustering works as follows:

GMM assumes data points are generated from a mixture of Gaussian distribution. Assume there are K Gaussian distribution, the mixture distribution can be defined as:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k) \tag{1}$$

π_k represents the probability that a data point belongs to the kth Gaussian component (cluster). μ_k and Σ_k correspond to the mean and covariance of the kth Gaussian component. π_k satisfies $0 \leq \pi_k \leq 1$ and $\sum_k \pi_k = 1$ according to the definition of probability. Given a data point $\mathbf{x}_i$, we consider there is a latent variable $\mathbf{z}_i$, corresponding to this data point that shows which Gaussian component it belongs to (it can also be interpreted as the data point's cluster identity). We model $\mathbf{z}_i$ as a binary vector of size K. If $\mathbf{x}_i$ belongs to cluster k, then only the kth value in $\mathbf{z}_i$ is 1 and all other values are 0s. For example:

$$\begin{aligned} \mathbf{z}_1 &= [1, 0, \cdots, 0] \\ \mathbf{z}_2 &= [0, 1, \cdots, 0] \\ &\vdots \end{aligned} \tag{2}$$

Given data $\mathbf{X}$, we need to infer $\mathbf{z}$ for all the data points, while the model parameters π_k, μ_k and Σ_k are unknown. This can be achieved with an EM algorithm.

Given initialized parameters, the EM algorithm conducts two steps:

- E step: With the current parameters π_k, μ_k and Σ_k, calculate the posterior distribution $p(\mathbf{z}_i|\mathbf{x}_i)$ for any i.
- M step: Given the $p(\mathbf{z}_i|\mathbf{x}_i)$ calculated from the E step, estimate parameters π_k, μ_k and Σ_k for all ks by maximizing the expected log-likelihood function.

(a). Denote the kth element in vector $\mathbf{z}_i$ by $\mathbf{z}_i(k)$. Please calculate the posterior distribution $p(\mathbf{z}_i(k) = 1|\mathbf{x}_i)$ in E step. Your solution can include the Gaussian distribution function $\mathcal{N}(\mathbf{x}_i|\mu_k, \Sigma_k)$. (Hint: use Bayesian formula)

Solution:

$$\begin{aligned} p(z_i(k) = 1|x_i) &= \frac{p(z_i(k) = 1)p(x_i|z_i(k) = 1)}{\sum_{j=1|K} p(z_i(j) = 1)p(x_i|z_i(j) = 1)} \\ &= \frac{\pi_k \mathcal{N}(x_i|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_i|\mu_j, \Sigma_j)} \end{aligned} \tag{3}$$

b. Assume that $p(\mathbf{z}_i(k) = 1|\mathbf{x}_i)$ is given, please calculate μ_k that maximizes the expected log-likelihood function:

$$\mu_k = \arg\max_{\mu_k} \sum_{i=1}^{N} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i|\mu_k, \Sigma_k) \right) \tag{4}$$

Solution: First calculate the derivative of the likelihood function w.r.t μ_k:

$$\frac{\partial L}{\partial \mu_k} = \sum_{i=1}^{N} \left(\frac{\pi_k \mathcal{N}(\mathbf{x}_i|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_i|\mu_j, \Sigma_j)} \cdot -\Sigma_k^{-1}(\mathbf{x}_i - \mu_k) \right) \tag{5}$$

By setting the derivative as 0, we can calculate μ_k

$$\mu_k = \frac{\sum_{i=1}^{N} p(\mathbf{z}_i(k) = 1|\mathbf{x}_i)\mathbf{x}_i}{\sum_{i=1}^{N} p(\mathbf{z}_i(k) = 1|\mathbf{x}_i)} \tag{6}$$

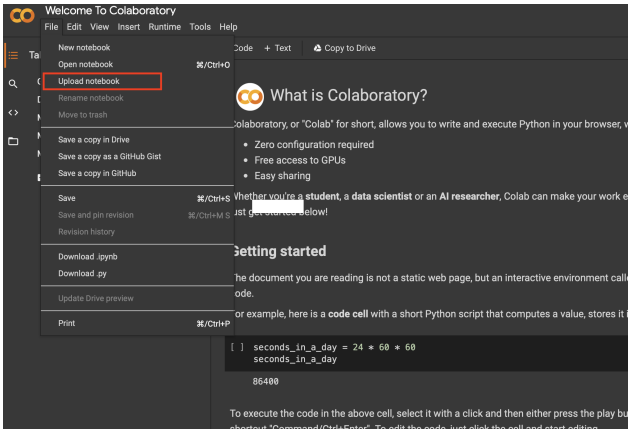
2 Programming: Comparison of different clustering algorithms [15 pts]

In our lecture, we talked about K-means clustering, spectral clustering and Leiden clustering. In this homework, we will practice with those three algorithms on different datasets, and see how those algorithms perform in different circumstances. You will need to complete the coding of the jupyter-notebook `hw3_1.ipynb`, which is in `hw3.zip` on canvas.

After you finish the code, briefly discuss the result that you obtain. You will also need to discuss the advantages and disadvantages of those three algorithms in terms of their clustering performance and running time. Please include your discussion in the pdf file that you submit. You will also need to submit `hw3_1.ipynb`. Please wrap it up with your notebook of the next problem into a zip file, and submit the zip file along with your pdf solution.

Note: We strongly recommend you to use google `Colab` for the programming, as we will also test your code using google `Colab`.

`Colab` is a free online version of the jupyter-notebook. You can learn more about `Colab` through the link: https://colab.research.google.com/notebooks/intro.ipynb. You can upload your `.ipynb` file onto `Colab` by clicking "Upload notebook" in "file" tab(Figure below).

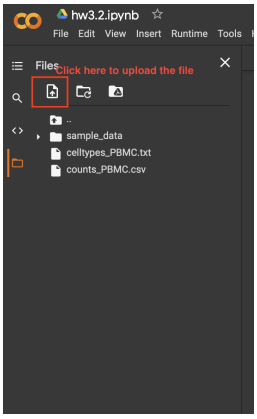


3 Programming: Autoencoder [15 pts]

In this part, you will use autoencoder to perform dimensionality reduction on a human peripheral blood mononuclear cells (PBMCs) dataset [1]. Please complete the code in `hw3_2.ipynb`. You will need to use `Pytorch` when you implement the autoencoder. Google `Colab` has the pytorch package already installed for you. You can also install `Pytorch` and run the code on your local machine. You may find `Pytorch` tutorial(https://pytorch.org/tutorials/) a useful reference.

You will need to submit `hw3_2.ipynb`. Please zip it with `hw3_1.ipynb`, and submit along with the pdf file to canvas.

Note: You may need to load the data onto Google `Colab`, you can do this by clicking upload file in the `Colab` interface:



References

[1] Hyun Min Kang, Meena Subramaniam, Sasha Targ, Michelle Nguyen, Lenka Maliskova, Elizabeth McCarthy, Eunice Wan, Simon Wong, Lauren Byrnes, Cristina M Lanata, Rachel E Gate, Sara Mostafavi, Alexander Marson, Noah Zaitlen, Lindsey A Criswell, and Chun Jimmie Ye. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nat. Biotechnol.*, 36(1):89–94, January 2018.