COGS 118B [Wi22]
Project 1 description

The aim of this project is to implement some unsupervised learning algorithms that you have learned so far. We will use the MNIST dataset, which consists of images of handwritten digits. We want to see how much information we can gain about the dataset using the methods learned in class, assuming that you have no access to labels. We have provided a notebook to you, where you should complete the desired code blocks and lines.

1. In **Section 1**, you will fit a **single Multivariate Bernoulli Distribution** to the dataset. Complete the code in **Section 1** of the notebook to get a Maximum Likelihood Estimate of the parameter for the Multivariate Bernoulli Distribution.

2. In **Section 2**, you will apply **Kmeans** to the given dataset to cluster the images. For this, complete the functions **calcSqDistances, determineRnk, recalcMus, runKMeans** as you did in HW3 Q4. The **runKMeans** function takes as input the number of clusters that you want to fit. For the purposes of this project, we are going to try with K = 10 and K = 20 clusters. After obtaining the locations of clusters for K = 10 and K = 20 clusters, execute the function **get_cluster_plot** to get an idea of how your clusters look. Provide a qualitative assessment of your results for K = 10 and K = 20 clusters.

3. In **Section 3**, you will fit a **Mixture of Multivariate Bernoulli Models** on the dataset. A brief review of the EM steps for this model is provided in the notebook and the update equations are also provided. You need to implement these equations to train your model using EM. Complete the function **train_EM_MOB** that takes as arguments **x : dataset, pi : initialization of mixing probabilities(π_k), theta : initialization of cluster locations (θ_k)** and **K: number of mixtures in the model**. For this project, limit yourself to a mixture model of 10 clusters only. Complete the code blocks for initialization of the parameters pi, theta and call the **train_EM_MOB** first on a small-sized dataset to ensure everything is working properly. Debug if needed. Once you get it working correctly, train the model on the full size of the dataset provided. Execute the **get_cluster_plot** function to plot the clusters that you got using this model. Provide a qualitative assessment of the performance and compare your plot to the case with Kmeans and Single Bernoulli Model.

PS:

1. See the hint given in notebook about vectorization in **train_EM_MOB** function. Avoid the use of nested for loops in your implementation otherwise the time taken will be significantly larger. Feel free to search the internet about methods of vectorization. Consult the TA's if you have any doubts. Good luck!

2. You can run the notebook in Google Collab. If you do want to run it in your local machine, a yaml file has been provided to you named **cogs118b.yaml**
Assuming you all have Anaconda installed, create a conda environment using provided file and commands in terminal as follows:-
conda env create -f cogs118b.yaml

This will take some time to install. After it is complete, activate your environment using:
conda activate cogs118b

Then, from this environment, launch your notebook.

If you face any problems with the above, please email **Sambaran (sghosal@ucsd.edu)**

SUBMISSION DETAILS:

You should work in groups of 2 or 3 people. Please use Campuswire to find group members if needed. Each group must produce its own code and figures. Copying code from other groups is a violation of academic integrity. If you have any questions or need help, please use Campuswire so that your question is visible (and can be helpful) to everyone.

Each group should prepare one submission including:

1. Your Python notebook with all your code. The notebook should include all of your analyses. It should be self-contained – i.e., anyone running your notebook should be able to reproduce all of your results.
2. Your code in a .zip file (only if you have additional code files besides the notebook)
3. A 2-page report.

The report should contain the following sections:

- INTRODUCTION: A brief introduction describing what you are trying to do and why.
- METHODS: A brief section, organized into 3 subsections, each describing the methods used (e.g. hyperparameters, any details of the code implementation, etc.)
- RESULTS: Describe your results. In addition to some text, this section should be organized by figure. Please create 3 figures with the results of each section of the project: one for single-Bernoulli, one for K-means, and one for the Mixture-of-Bernoulli. Each figure can include multiple subpanels if needed, and each figure must include a caption describing any figure element. Please make sure to label all your axes.
- DISCUSSION: A brief discussion about what you found, what you learned with this project, and explaining any difficulties you have encountered.

GOOD LUCK!