Problem 4: Yeast Colonies [Email Submission]

By Henry Tang

1 Unsupervised Clustering (10 pts)

Sally is a Nobel Prize-winning mycologist at Princeton. Recently, she's been studying three new fascinating yeast colonies that she discovered in the wild. The types are creatively called Label 0, Label 1, and Label 2. She picked up a number of samples and brought them back to her lab, where she studied their various growth patterns. After collecting the data, she went home where she spent some fun time solving large Rubik's Cubes. The next day when she arrived, she discovered to her horror that a naughty student who despises her Rubik's Cube solving abilities had ripped off the stickers identifying the type of the various samples. In addition, the yeast samples themselves had been destroyed. She still has data detailing 5 characteristics of each of the yeast samples, but she is now unable to determine which type of yeast each sample is. Can you help her out?

Setup: For this problem, you will be using Google Colab, which is an intuitive way of running machine learning code in Python. *Make sure you're signed into your Princeton Gmail Account to access Colab.* You can find the starter notebook in Python here (which we *highly recommend*), or a Java version here (only use this only if you know no Python at all). Follow the instructions in the notebook and the training dataset will be automatically downloaded. You will see that there are 640 total data points/yeast samples. In python, each row of the dataset which will be in the form of a pandas dataframe, representing a single yeast sample. In Java, you will have a 2D-list where each row again represents a data point.

Your Goal: Predict the labels for each data point in the training set and output a text file titled *Yeast-Part1.txt*. Each of the 640 lines should contain a single integer prediction class for the yeast type of that data point (0, 1, or 2). The order of the data points should be the same as the order they were provided to you in the original `csv` file (this corresponds to the same order that the data appears once imported). We have provided you with an example output file (with randomly generated labels) here for you to look at.

Important Note: Once you separate the data points into three distinct classes, you'll notice that you still don't know which of the three classes correspond to Label 0, Label 1, and Label 2. Do not worry about this. When we test your code, we will test it against all permutations and grade you based on the one with the highest accuracy. In other words, if we're only testing on the first five data points, then a class prediction of (0,0,1,0,2) would yield the same accuracy in our testing script as (2,2,0,2,1) since the class labels are permutations of each other ($0 \to 2, 1 \to 0, 2 \to 1$).

Some Helpful Background: Supervised vs. Unsupervised Learning

In machine learning, there are two main classes of problems: supervised and unsupervised. In *supervised* learning, you are given some data with characteristics (also called features), as well as their labels (you can think of these as a "classification" of sorts). You would then want to train a model that given input characteristics, can accurately predict the label of never seen before data.

In this problem, however, we're going to be working with an *unsupervised* learning problem. Given characteristics of data, we want to figure out what label/group each data point belongs to. But wait, you may ask, how is this possible? One method is to look for similar characteristics among all data points. For instance, in this problem, since there are 5 characteristics for each data point, we can treat each data point as a 5-dimensional vector in $\mathbb{R}^5$. Then, we can imagine choosing a set of three special points (which we can call centroids), also in $\mathbb{R}^5$, and associating each centroid with a yeast label. For each data point, we can classify it as belonging to the yeast class whose centroid is closest to it.

Now, the question remains how we can find these centers? One method is to k-means. For more information, see here. Keep in mind though that for better scores, you might want to consider other unsupervised learning methods. Another important question is to evaluate how well your k-means is performing. One way to do so is to compare its results against other unsupervised algorithms, and check to make sure the resulting predictions are similar.

Grading and How to Submit

Email the text file containing your predicted labels to coscon.written.submission@gmail.com with subject line *Problem4aSubmission* and file name *YeastPart1.txt*. If you must resubmit, *respond to the thread where you sent your original submission; we cannot guarantee that your resubmission will be graded otherwise.*

Your text file will be evaluated for accuracy against the true yeast labels for Sally's dataset. If the resulting accuracy in percent is a, then your score for Part 1 is $\min(10, (a - 40) * 0.2)$.

2 Further Label Prediction (5 points)

Charlotte, Sally's coworker, has also been investigating these three types of yeast. She has analyzed her own samples (and recorded down exactly the same five attributes that Sally analyzed), and is now asking for Sally's help in determining what type of yeast these data samples correspond to. Using the labels you obtained in Part 1, can you help Sally and Charlotte out by predicting the labels of these new yeast colonies?

Instructions: You can find the starter notebook in Python here and the starter notebook in Java here. *You must use a Colab Notebook for this problem.* You will also not have access to Charlotte's dataset, as your model should work regardless of what it is! Your Colab code must generate a single text file titled $YeastPart2.txt$ into the local directory. The format of this text file should be the same as in Part 1, which the predicted label (0, 1, or 2) for each data point presented one per line.

Grading and How to Submit

Email the Colab Notebook above to coscon.written.submission@gmail.com with file name *Part2.ipynb* and subject line *Problem4bSubmission*. In addition, in your email, you are permitted, but not required, to submit up to two additional files that may be needed in your model (such as a saved model parameters file). We will be running your code ourselves on test servers after the contest on the test (Charlotte's) dataset. All the files you sent will be put into the same folder. The absolute path specified in the fourth cell of the Colab notebook will be modified accordingly by us when grading, but try to make it as easy as possible for us to run your model :)

Your text file will be evaluated for accuracy against the true yeast labels of Charlotte's dataset. If the resulting accuracy in percent is a, then your score for Part 2 is $\min(5, (a - 45) * 0.1)$.

Final helpful hints: You may want to consider using any of the following Python packages: `torch 1.9.0`, `scikit-learn 1.0.1`, `tensorflow 2.7.0`, `numpy`, `pandas`, and `joblib 1.0.1`. If you do, though, please try to make sure your code works with the versions that we've specified. Yet another reason you should try to use Python if you can! And come to office hours if anything is unclear, Colab isn't working, you have no idea how to get started (on either part) or you just want to learn more!