CSci 4695 and 6965: AI for Conservation
Fall Semester, 2023
Homework 2
Due: Friday, September 15, 2023 at 11:59:59pm

Detailed submission instructions and example data will be posted on-line by the end of the day on Monday September 12.

Homework Guidelines

Your homework submissions will be graded on (a) correctness of your solution, (b) clarity of your algorithms and code, and (c) effective use of the methods we are studying. When a solution asks for analysis of your results, it is important that you generate clear insights.

Submission Guidelines — Jupyter Notebook

Your solution **must be** uploaded to Submitty as a single Jupyter notebook for the entire assignment. Each problem solution must be contained in a single notebook cell, with all imports, functions, classes, and testing code. Assume all input files needed are in the same directory as the notebook. Submit the notebook with all results of your results completely run and embedded. Be aware, however, that I am likely to re-run at least one or a few cells of your notebook during grading.

Academic Integrity

You are strongly encouraged to make use of any and all on-line resources you can find to help you understand the problems and the methods you choose to use to solve them. The code you write for your solution should be your own, however. Copying or sharing solutions from other students is not acceptable.

Problems

These problems are focused on developing your skills in working with vector programming, NumPy and images. **If you already have these skills and especially if you are in the graduate version of the class, please jump ahead to the Advanced section at the end of this document.**

1. **(15 points)** Write a Python script to concatenate two images into a single output. If the images have different heights, shrink the image with the greater height so that it matches the smaller height. For example, if the first image has $h_1, w_1 = 512, 400$ and the second image has $h_2, w_2 = 640, 480$ then the second image should be resized to be $h_2, w_2 = 512, 384$, and the concatenated image should have height 512 and width $480 + 384 = 864$. Please output the dimensions of the final image and display it.

2. **(15 points)** Write a Python function that computes and outputs the mean and standard deviation of each of the red, green and blue values in a color image. The output should be just three lines, for example,

```
red: 129.1 12.6
green: 197.2 40.3
blue: 204.4 23.5
```

Please note that the output should include only one decimal place of precision.

3. **(25 points)** The *retrieval* task is a variation on the classification task where for each query q a database is searched and possible matches that could be from the same category as q are returned in order of likelihood. Referring to these matches as $d_1, d_2, d_3, \ldots$, we say that the performance on the query has rank k if d_k is the first match from the same category as q, in other words if $d_1, \ldots, d_{k-1}$ are each from a different category as q.

   Suppose we run a series of queries from different categories and we would like to summarize the results. For each category we'd like to know the average of k. We'd also like to know the average across all categories. This could be either the average of the averages, which is called the *macro average*, or it could be the overall average across all queries, which is called the *micro average*. The macro and micro averages are only guaranteed to be the same if we have the same number of queries for each category.

   Suppose we represent the query results in a two-dimensional array $\mathbf{Q}$, where row i of $\mathbf{Q}$ corresponds to the i-th category and entry j of row i is the value of k for the j-th query for category i. Note that if $\mathbf{Q}_{i,j} = -1$ then the j-th query should be ignored.

   Output the average k for each category and then output the macro average and the micro average. All values should be acccurate to 2 decimal places. The only `for` loops should be at the end to output the results.

   Here's a simple example to illustrate with two categories

$$\mathbf{Q} = \begin{pmatrix} 2 & -1 & 5 & 1 & 4 & 4 \\ 4 & 8 & -1 & -1 & -1 & 11 \end{pmatrix}$$

```
Per query
0: 3.20
1: 7.67
Macro: 5.30
Micro: 4.50
```

   Finally (and briefly), why might we be interested in both the macro and micro averages?

4. **(20 points)** In class, we created a binary image from a grayscale image using a fixed threshold of 128. In this problem I'd like you to create an image that has four values: 0, 85, 170 and 255, and there should be approximately the same number of pixels with each value. To do this, pixels assigned the value 0 should be the darkest 25% of the pixels, in other words, the pixels whose intensities are in the lowest quartile of the image intensities. Pixels assigned the value 85 should have intensities in the 2nd quartile. Pixels assigned the value 170 should be from the 3rd quartile. Finally, pixels assigned the value 255 should be in the 4th quartile of intensity — i.e. the brightest 25% of the pixels.

   Output the three breaks between the quartiles. These breaks are the smallest intensities mapped to the second, third and fourth quartile. Then output the final image. As a simple example, the "image"

$$\begin{pmatrix} 15 & 128 & 199 & 75 \\ 240 & 195 & 129 & 233 \\ 91 & 234 & 65 & 145 \end{pmatrix}$$

   should produce the output:

```
Quartile breaks: 91, 145, 233
```

and

$$\begin{pmatrix} 0 & 85 & 170 & 0 \\ 255 & 170 & 85 & 255 \\ 85 & 255 & 0 & 170 \end{pmatrix}$$

For Advanced Students Who Already Know NumPy and PyTorch

My current plan is to make this required for students registered for the graduate version of the class. Please let me know if you feel differently.

We are creating this course from scratch — meaning that there is a lot of groundwork needed — and I need some help. Over the next month we will be covering neural networks, PyTorch, animal species id, and plant species distributions. We'll need

1. References for discussion of application problem formulations and social context.
2. References for discussion of technical problem definition and solutions.
3. Problem definitions that can be addressed in class and in homework assignments. Preferrably these should start easy and gradually increase in length and difficulty.
4. Data sets appropriate for work on these problem definitions, properly cleaned (unless cleaning itself is proposed as part of the assignment).
5. Demonstrations of advanced / publication-level solutions, both in terms of code and results.

What I need specifically by next Friday from each advanced student:

1. Email to me telling me you are joining this effort.
2. Preference of problem areas: how much would you prefer to study species distributions for plants vs. animals?
3. References to descriptive resources
4. References to data sets
5. Sketch of potential problems for class and homework.