Intro to ML and Big Data
Assignment 4 - K-means

CS 4731 — Dr. Mihail

Department of Computer Science

Valdosta State University

October 22, 2019

Do not attempt a Mihail homework the night it's due.

1   Introduction

Images stored digitally typically allow a pixel to have 24 bit colors (8 bits for red, green and blue), for a total of roughly 16 million possible colors. Much of this information is reduntant, since similar colors are difficult to perceive by the human eye. Therefore, one possible image compression technique is to reduce the total number of colors.

In this assignment, you will implement the K-means algorithm in Python, and apply it to reduce the number of colors in images. Here, the data consists of pixel colors, each pixel is treated indepenently, irrespective of its spatial position. Color images have pixels with three channels (red, green and blue). Performing K-means on the set of all pixels in an image for some K results cluster centers that correspond to colors. You will then replace the colors in the original image with the color of the nearest cluster center.

The choices of K you will implement are 2, 4, and 8. These correspond to an indexed color scheme, where each pixel is represented by 1, 2 and 3 bits, respectively. They index into a color palette, stored separately. The color palette will be given by the cluster centers.

2   Dataset

For this project you may use any online image with an accessible URL. The skimage.io library has a function called *imread* that loads an image, given a URL. For example:

```
from skimage.io import imread
im = imread('http://www.youandthemat.com
            /wp-content/uploads/nature-2-26-17.jpg')
```

will load the first image in the examples show below.

3   Technical considerations

- Since some of the information in imagery is redundant, you don't have to use all the pixels in the image.

- Make sure your images are normalized in the $[0, 1]$ range.

4   Project requirements

You have to use the implementation given in the class slides. You may NOT use sklearn or any other implementation. Your code must generate the high resolution equivalent plots as shown in Figure 1. The URLs are as follows:

1. `https://s3-us-west-2.amazonaws.com/uw-s3-cdn/wp-content/uploads/sites/6/2017/11/04133712/waterfall.jpg`

2. `http://www.youandthemat.com/wp-content/uploads/nature-2-26-17.jpg`

3. `https://buddhisteconomics.net/wp-content/uploads/2017/10/hdwp693968124.jpg`

4. `https://upload.wikimedia.org/wikipedia/commons/d/dc/Skyscrapers_of_Shinjuku_2009_January_%28revised%29.jpg`

5. `https://www.worldatlas.com/r/w728-h425-c728x425/upload/db/04/18/toronto-ontario.jpg`

6. `https://www.valdosta.edu/academics/online-programs/images/slideshow-new-vsu-arch.jpg`

7. `https://www.valdosta.edu/administration/planning/images/planningimage.jpg`

8. `http://www.carterusa.com/project-images/develop/program-Project-Management/higher-education/Valdosta-State-Student-Housing/gal-1.JPG`

- **Write-up** You will write-up your project in a LaTeX document with all the decision points along the way documented and discussed. This has to have a section titled "What I've learned" where you will discuss what you learned.

5   Due date

The project is due before the end of the day Sunday, October 27th.

6   Example results

Figure 1: Sample results from a successful implementation.