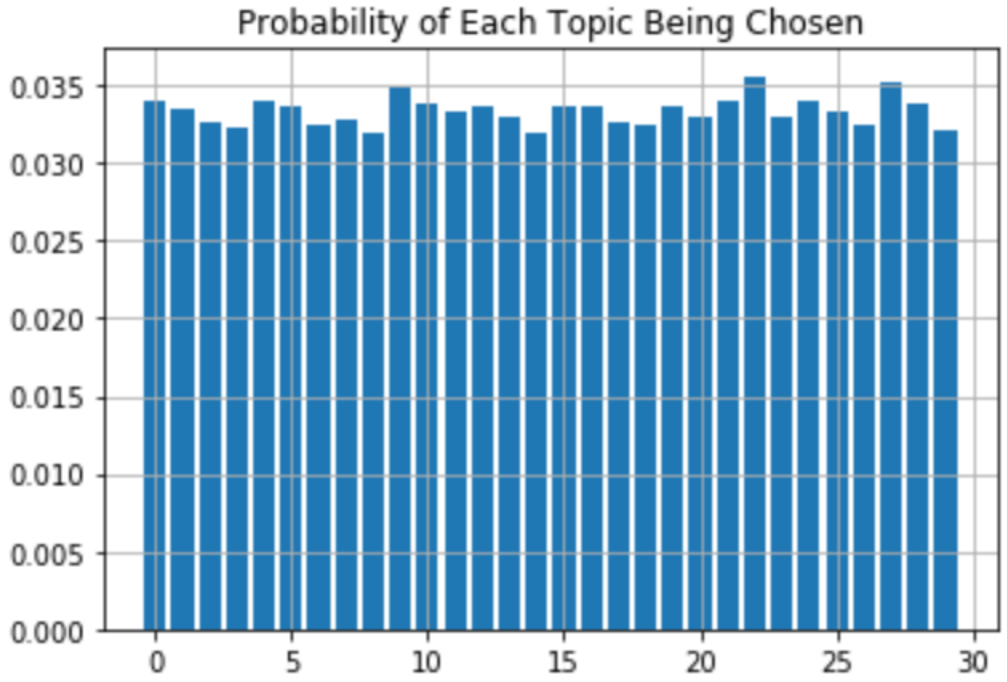
Problem 1

```
Convergence criteria: norm(mu_old-mu_new) < 0.001
** Convergence progress screenshot in appendix
```




```
** We experienced some bugs trying to get the top words for each cluster **

Cluster - 0
['adelson'] ['advantage'] ['advocate'] ['advocate'] ['advocated'] ['advocated'] ['aerospace'] ['affect']
['afford'] ['africa']
----------------------
Cluster - 1
['adequately'] ['adopted'] ['advan'] ['advanced'] ['afford'] ['aftereffect'] ['against'] ['agc'] ['age']
['ages']
----------------------
Cluster - 2
['adjusting'] ['admissibility'] ['adopt'] ['advantageous'] ['advocate'] ['aea'] ['aero'] ['aeronautic']
['aerospace'] ['afterward']
----------------------
Cluster - 3
['aero'] ['aerospace'] ['affect'] ['affected'] ['africa'] ['aged'] ['agencies'] ['aggregated'] ['aggregating']
['aggregation']
----------------------
Cluster - 4
['adopting'] ['adult'] ['adversary'] ['aerial'] ['aertsen'] ['aertsen'] ['afterward'] ['against'] ['aged']
['agglomerative']
----------------------
Cluster - 5
['adaptability'] ['advances'] ['aer'] ['aero'] ['afford'] ['aftereffect'] ['afterward'] ['aged'] ['agency']
['agent']
----------------------
Cluster - 6
['adjacent'] ['aea'] ['aertsen'] ['aertsen'] ['agencies'] ['agency'] ['agent'] ['aggressive'] ['agnostic']
['agonist']
----------------------
Cluster - 7
['adder'] ['admission'] ['adv'] ['advanced'] ['advice'] ['aero'] ['aeronautic'] ['afterward'] ['against']
['agation']
----------------------
Cluster - 8
```

```
['adjusted'] ['adopted'] ['advantageous'] ['adverse'] ['advice'] ['advocate'] ['aerospace'] ['afosr'] ['agc']
['agencies']
-----------------------
Cluster - 9
['adopted'] ['adv'] ['aea'] ['aerial'] ['affect'] ['affine'] ['against'] ['agencies'] ['ages'] ['agglomerative']
-----------------------
Cluster - 10
['adult'] ['aeronautic'] ['afosr'] ['africa'] ['aftereffect'] ['aftereffect'] ['agency'] ['ages'] ['ages']
['aging']
-----------------------
Cluster - 11
['adaptation'] ['adopt'] ['adopt'] ['advanced'] ['afforded'] ['afterward'] ['aggregation'] ['aggregation']
['aggressive'] ['ago']
-----------------------
Cluster - 12
['advocate'] ['aea'] ['aerospace'] ['affine'] ['afosr'] ['aftereffect'] ['agation'] ['agc'] ['age'] ['agency']
-----------------------
Cluster - 13
['adaptability'] ['adaptable'] ['advan'] ['advocated'] ['aerial'] ['aero'] ['aero'] ['aeronautic'] ['affect']
['against']
-----------------------
Cluster - 14
['admissibility'] ['advance'] ['advanced'] ['afterward'] ['against'] ['against'] ['against'] ['agation'] ['age']
['agent']
-----------------------
Cluster - 15
['aer'] ['aerial'] ['africa'] ['age'] ['ages'] ['ages'] ['aggregate'] ['aggregate'] ['aggregated']
['aggregating']
-----------------------
Cluster - 16
['adjustable'] ['advance'] ['africa'] ['aged'] ['agencies'] ['aggregate'] ['aggregation'] ['aggressive']
['aggressive'] ['aging']
-----------------------
Cluster - 17
['adaptability'] ['admissibility'] ['advance'] ['advocate'] ['affect'] ['affected'] ['afferent'] ['afford']
['agency'] ['aggregate']
-----------------------
Cluster - 18
['admit'] ['admit'] ['adopt'] ['advanced'] ['adverse'] ['advice'] ['aea'] ['aero'] ['afforded'] ['against']
-----------------------
Cluster - 19
['adjusting'] ['advantages'] ['advice'] ['affine'] ['afforded'] ['afforded'] ['agation'] ['aggregate']
['aggregated'] ['aggregating']
-----------------------
```

Problem 2

The second problem in Homework 7 asks the student to perform clustering on the pixels contained within images provided by the instructors, and then reassign the color of each pixel to the mean RGB value of that pixel's cluster. After this color replacement, the resulting image approximates the original image that's been reduced from the original range of pixel colors down to only 10, 20, or 50 colors depending on the value specified for k.

To initialize our cluster centers and cluster assignments, we have chosen to use the scikit-learn.kmeans library to perform an initial clustering and predictions. Having obtained this initial information, we are able to manually calculate the initial weight of each cluster calculating the percentage of pixels that belong to each cluster.

After initialization of our values with Kmeans, we used an Expectation-Maximization algorithm to further refine our cluster centers and weights and improve the reduced image that is produced. The E-step of the algorithm assigns new probabilities that each data point belongs to each cluster. The output of the E-step acts as an input to the M-Step and allows us to calculate new cluster centers and new cluster waits.

During the M-Step, we use logarithmic values when calculating new values for cluster weights to help with numeric stability.

To test convergence our EM algorithm, after every iteration we measured the total linear distance between the new cluster centers and the previous cluster centers and compare that value to a minimum threshold of 0.1. We've also

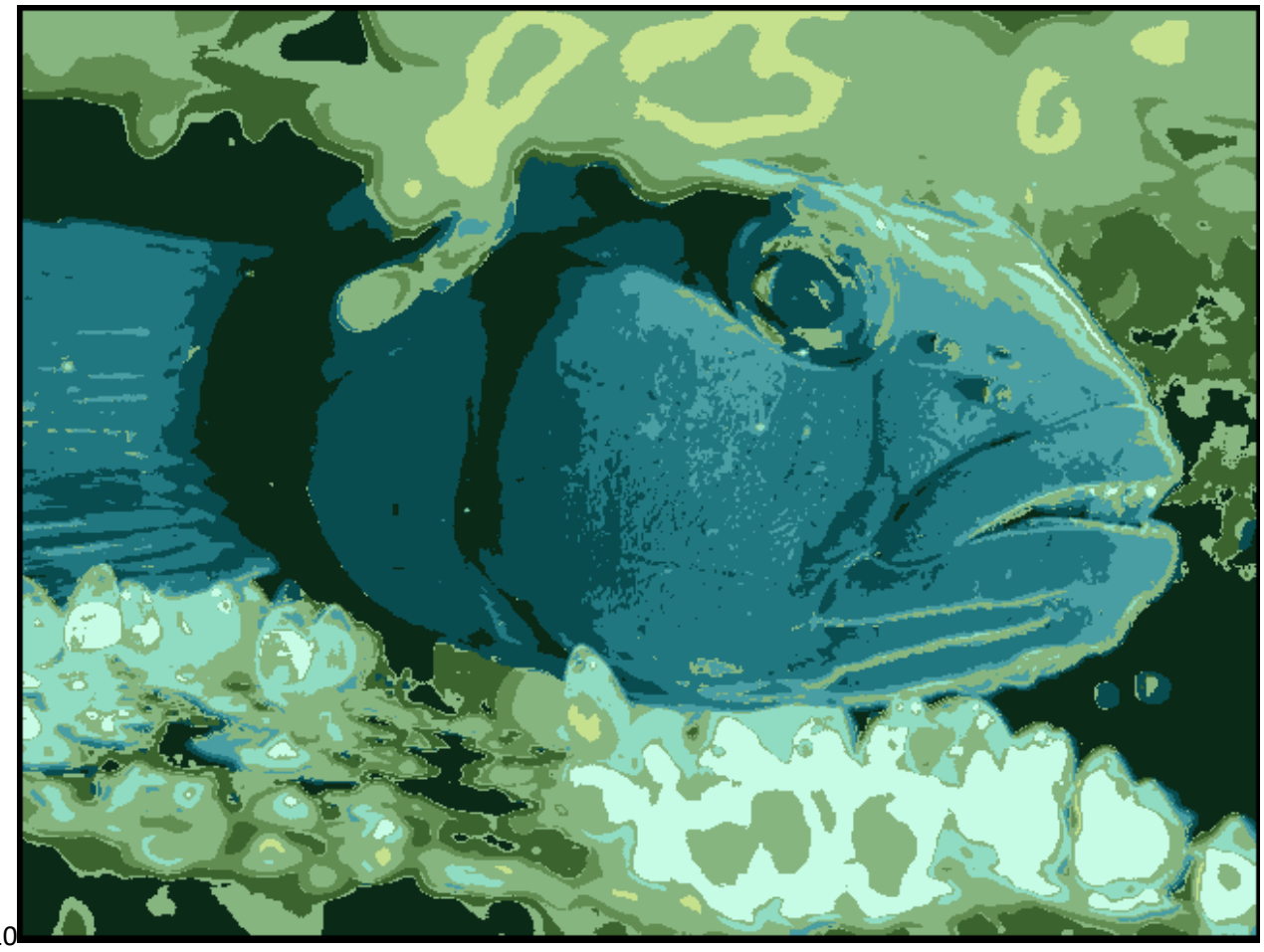
limited the maximum number of iterations to 100, though during our testing we have never had to enforce that iteration limitation. In general, the larger the value used for k, the more iterations our algorithm took to meet the minimum threshold for convergence. This intuitively makes sense because when more cluster centers are measured the sum of their distances is likely to grow as k grows.

Convergence of EM algorithm in Image Segmentation:

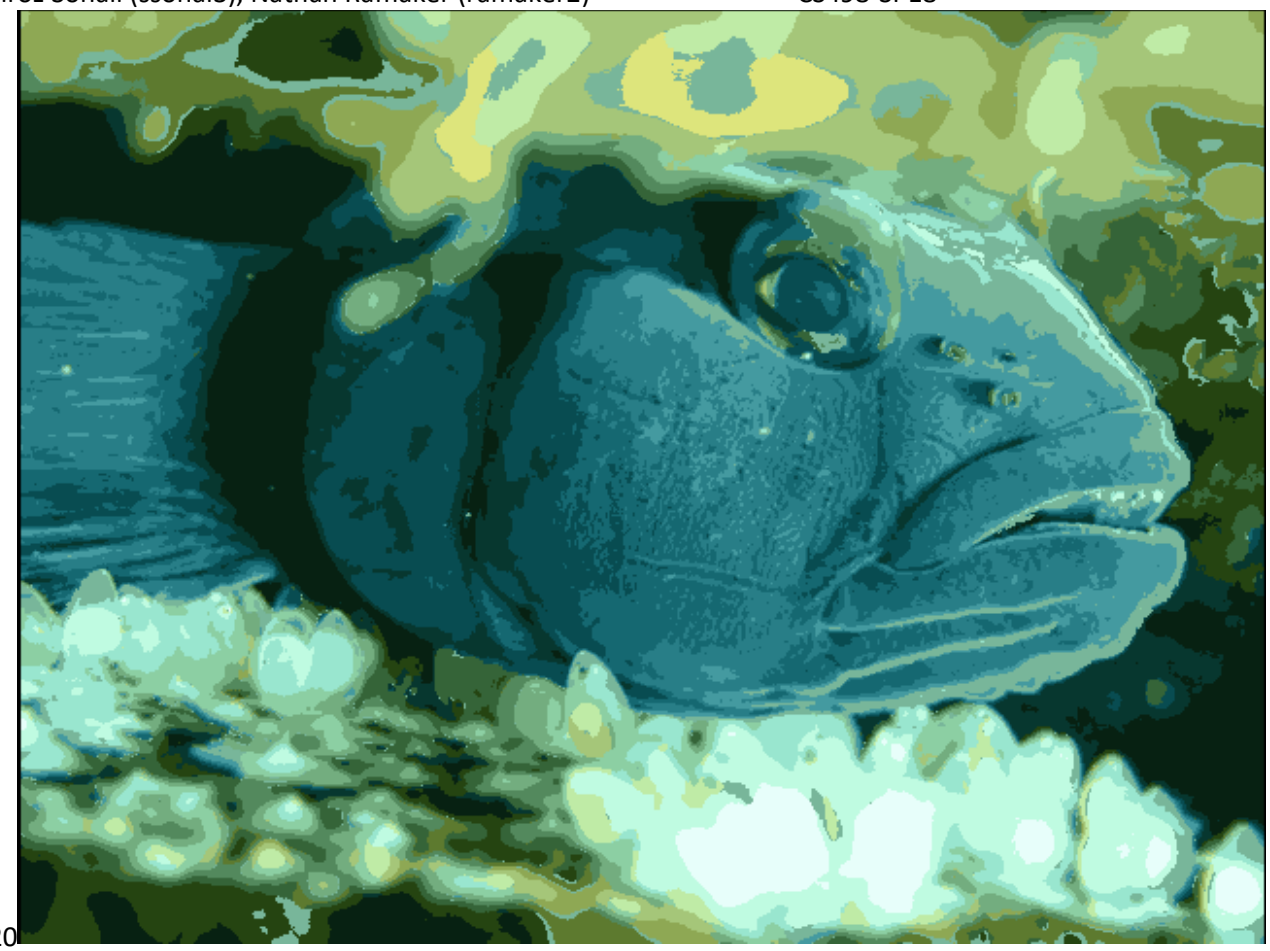
The effect of this color reduction is most apparent in areas of the images that contain gradients, such as the upper right-hand corner of the image RobertMixed03.jpg. This section of the original image has great color variation in the mentioned section, and as the image's pixels are reduced to a lower and lower number of clusters, the resulting image has greater difficulty reproducing that variation.

Image: RobertMixed03.jpg

EM clusters after K-Means initialization.



K=10



K=20

K=50

Image: smallstrelitzia.jpg

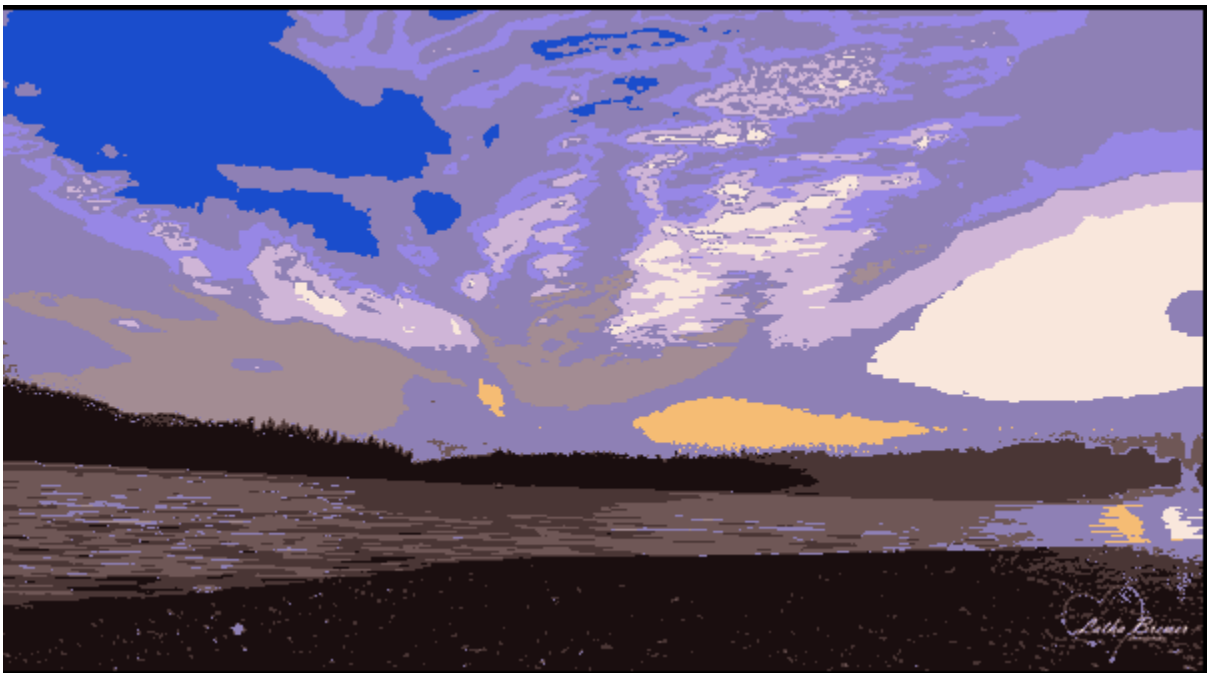
K=10

K=20

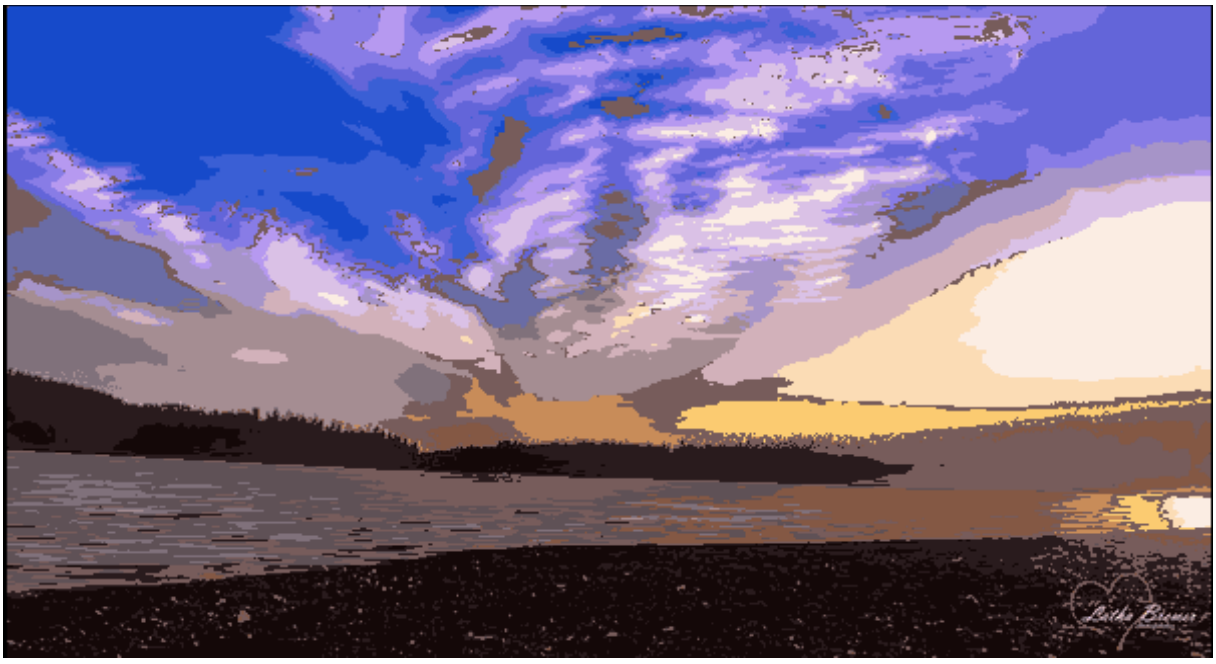
K=50

Image: smallsunset.jpg

K=10



K=20



K=50

Appendix:

Libraries used

PIL – (https://pypi.python.org/pypi/Pillow/2.7.0) – Used for Image processing

Numpy – (http://www.numpy.org/) – Used for matrix operations

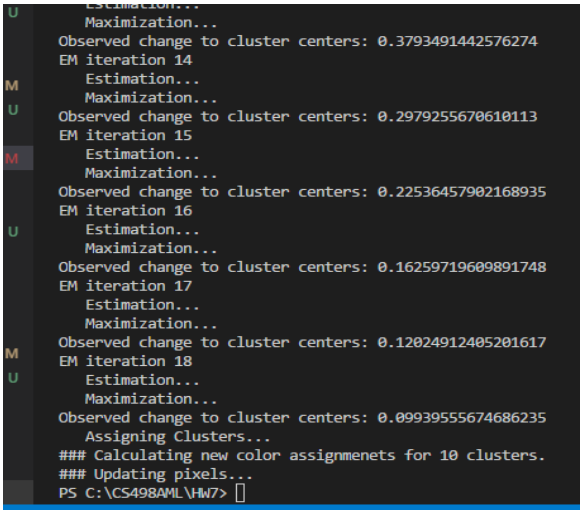
Scikit-learn – (http://scikit-learn.org/stable/) – Used for Kmeans initialization

Matplotlib – (https://pypi.python.org/pypi/matplotlib) – Used for Histograms

Pandas – (https://pandas.pydata.org/) – Used for loading and accessing data

Referenced Links:

- Guidance document from TAS for transitioning EM theory to code - https://piazza.com/class/jchzguhsowz6n9?cid=1253
- Code sample for EM algorithm - https://stats.stackexchange.com/questions/55132/em-algorithm-manually-implemented
- Thread and code samples for getting pixels from image with PIL- https://stackoverflow.com/questions/1109422/getting-list-of-pixel-values-from-pil
- Code sample for extracting column from an array - https://stackoverflow.com/questions/903853/how-do-you-extract-a-column-from-a-multi-dimensional-array
- Thread and code samples for calculating euclidean distances - https://stackoverflow.com/questions/1401712/how-can-the-euclidean-distance-be-calculated-with-numpy
- Example implementation of Gaussian EM in Python - https://github.com/mcdickenson/em-gaussian/blob/master/em-gaussian.py
- Code Sample for getting index of maximum value in array - https://stackoverflow.com/questions/268272/getting-key-with-maximum-value-in-dictionary



Convergence of image segmentation EM

```
##### HW7 Topic Modeling #####
### Performing E/M algorithm: iteration 0
   E-step
   M-Step
   Detected change of 104.33239830447722
### Performing E/M algorithm: iteration 1
   E-step
   M-Step
   Detected change of 0.46894132158417345
### Performing E/M algorithm: iteration 2
   E-step
   M-Step
   Detected change of 0.046487649914594795
### Performing E/M algorithm: iteration 3
   E-step
   M-Step
   Detected change of 0.004410369132571547
### Performing E/M algorithm: iteration 4
   E-step
   M-Step
   Detected change of 0.000420887245486301
```

Convergence of topic modeling EM