

1)

(e)

i. As I began to test the program, I started with setting `n_colors` to 20. That test didn't give me the noticeable results that I was expecting, but still managed to slightly blur the image. Then I set `n_colors` to 30, which gave me a clearer image of myself. Afterwards, I set `n_colors` to 2, which gave me the results I was expecting my first test. The image was completely red and black, which were the two most dominant colors in the original image. However, the red seemed to be off colored from the red in the original image. This lead me to think that it simply divided the colors in to lights and darks, and blended those instead. Instantly thinking setting `n_colors` equal to 64 would give me something close to my original image, I tested my hypothesis which was conclusive.

ii. This program could be used in various different ways. I noticed that many of the images looked very similar to old school filters. Which means with a little more tweaking to the code, it could be made to spot all areas of a certain color and set it to colors of your choice. Another significant application for this program could be image compression. Rather than saving the image with every color present, it would only select a fraction of the original amount, producing a slightly more blurry picture with a significant reduction in size.

iii. The resulting image was funny at the end because it was a less colorful image of me.

2)

(c)

```
Last login: Tue Nov 15 00:34:02 on ttys000
MacBook-Pro-2:~ RyanNeumann$ python aprt2
python: can't open file 'aprt2': [Errno 2] No such file or directory
MacBook-Pro-2:~ RyanNeumann$ python part2
Neurons 1, eta 0.1. Testing set CV score: -30.223334
Neurons 1, eta 0.3. Testing set CV score: -17.720566
Neurons 1, eta 0.4. Testing set CV score: -8.208284
Neurons 5, eta 1.0. Testing set CV score: -7.627916
Neurons 7, eta 0.6. Testing set CV score: -6.151499
/Library/Python/2.7/site-packages/sklearn/neural_network/multilayer_perceptron.py:563: ConvergenceWarning: Stochastic Optimizer: Maximum iterations reached and the optimization hasn't converged yet.
  % (), ConvergenceWarning)
Neurons 95, eta 0.1. Testing set CV score: -5.637143
Iteration 1, loss = 1.58852058
Iteration 2, loss = 0.41986356
Iteration 3, loss = 0.13627686
Iteration 4, loss = 0.12920106
Iteration 5, loss = 0.06827523
Iteration 6, loss = 0.03771268
Iteration 7, loss = 0.02518688
Iteration 8, loss = 0.02932599
Iteration 9, loss = 0.02090396
Iteration 10, loss = 0.01527530
Iteration 11, loss = 0.02129298
Iteration 12, loss = 0.02177219
Iteration 13, loss = 0.02622311
Training loss did not improve more than tol=0.000100 for two consecutive epochs. Stopping.
Training set score: 0.882575
Testing set score: 0.861835
MacBook-Pro-2:~ RyanNeumann$
```

(d)

Running the program for 1,000 samples, I received multiple outputs with the best number of neurons as well as the best learning parameter. As the number of Neurons increase, so did the Testing set CV score. I suppose with the increase in Neurons, the network performance will be better at performing complex solutions. However adding new structures to the network, with too much internal structure, will only slow down the network.