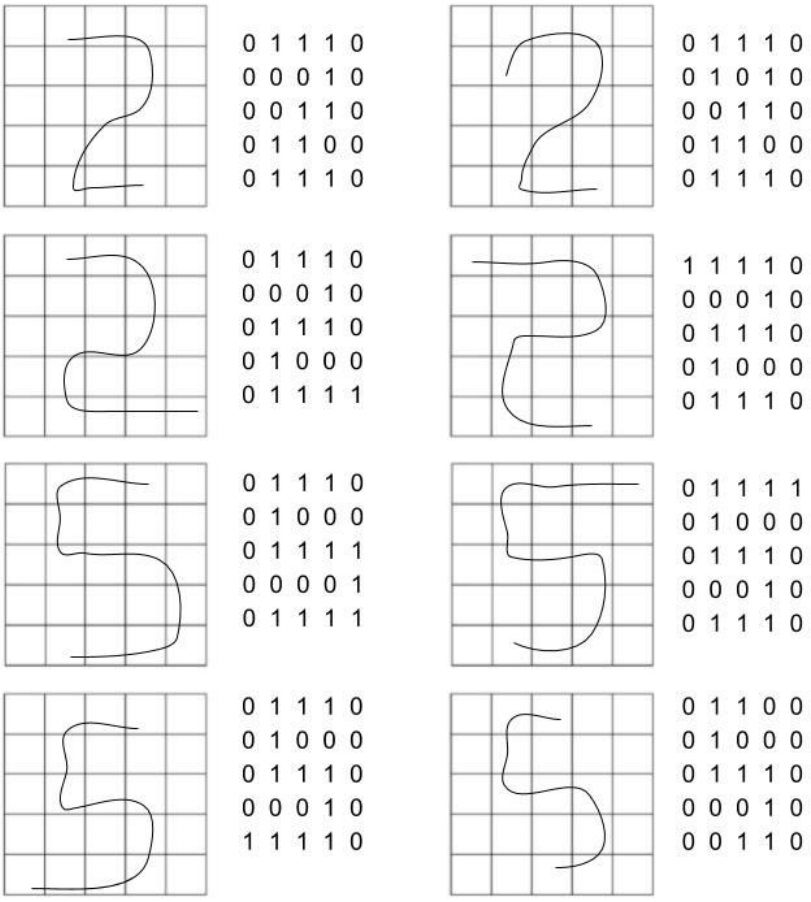
Calvin Zikakis
Artificial Intelligence
Assignment 4 Report
11/19/2019

Note Please Read:

I do not think my classification function is working properly (or maybe retrieve).

This report was written when using euclidean distance as the method to classify if the data was a two or a five. The code has been updated but is not running the classifier properly. I'm going to leave this report in terms of euclidean distance because it provided much more interesting results that provide a lot of insight. I think a semi-working classifier based off euclidean distance provides more education value than a non-working one. Please keep that in mind as you read this report.

Part 1: Data Preparation



Part 2: Hopfield Network

My network perfectly classified the digits provided resulting in an accuracy of 100%.

```
Label: two
Classified as:  two

Label: two
Classified as:  two

Label: two
Classified as:  two

Label: two
Classified as:  two
```

```
Label: five
Classified as:  five

Label: five
Classified as:  five

Label: five
Classified as:  five

Label: five
Classified as:  five
```

My network is not making errors because my testing data has very little variance to it compared to the training data. I designed my test data to have very slight differences than the training data because I knew that we would only be using 2 patterns for training. If my numbers were more complex, the classifier would not have worked as well and would have resulted in a high error rate. If one was to make the test data more complex then we would need a larger training group to bring the error rate up more.

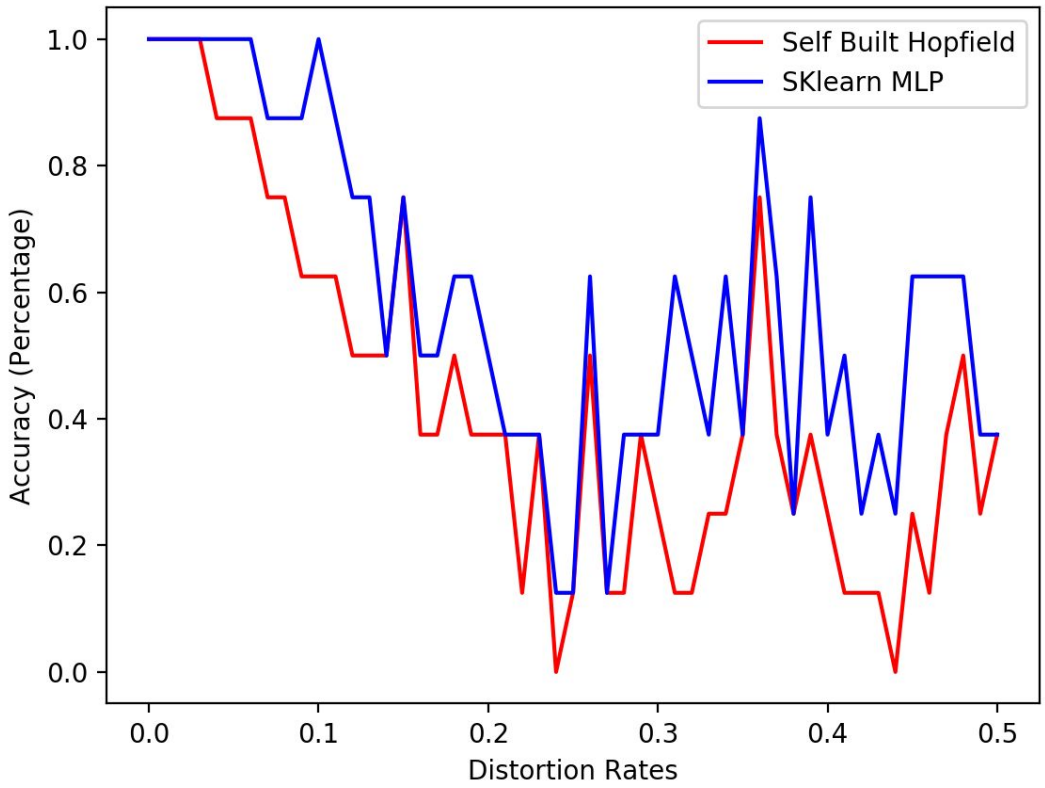
Part 3: Train a MLP

Once again we see a perfect classification resulting in an accuracy of 100%.

```
MLP Classifier
accuracy score =  1.0
```

The same explanation for part 2 applies here: My network is not making errors because my testing data has very little variance to it compared to the training data. I designed my test data to have very slight differences than the training data because I knew that we would only be using 2 patterns for training. If my numbers were more complex, the classifier would not have worked as well and would have resulted in a high error rate. If one was to make the test data more complex then we would need a larger training group to bring the error rate up more.

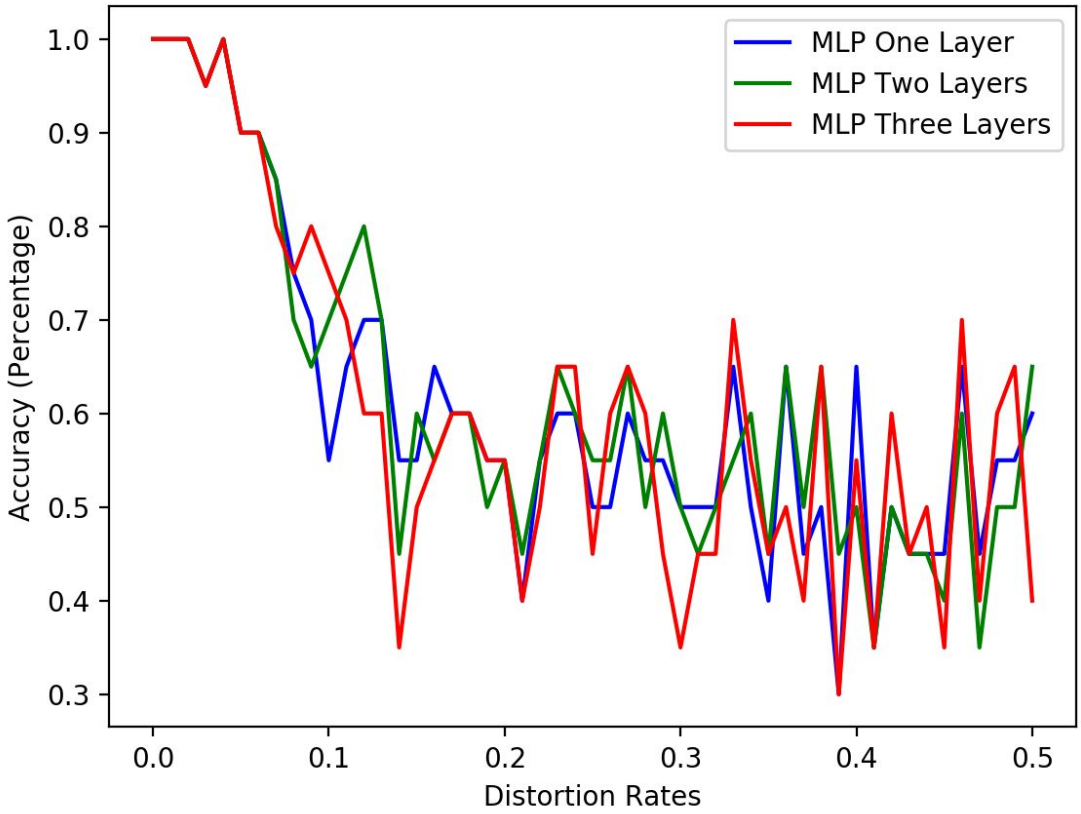
Part 4: Distortion



The above plot shows the accuracy of each classifier versus the distortion rate put into the classifier. Overall, the hopfield network does not perform as well as the MLP classifier. As the data distorts both classifiers suffer dramatically reducing the accuracy from 100% with no distortion to around 40% with a distortion value of 0.5. These values could easily be improved by providing more training data.

Because my testing data was so similar to the original data, when it distorts there is a higher likelihood of it disorting away from the original training data. If our test data had more variance to it, it would be more likely to converge towards the training data.

Part 5: Experimenting with number of hidden layers



As you can see in the above graph, the amount of hidden layers in the MLP classifiers did not make a difference. The data points are very much so varied and kinda random with their classifications after a certain point of distortion. I would like to see how these perform on a much larger dataset.