Brianna Shade

CS545: Machine Learning

3/9/14

Project 4: K-means Clustering Classifying Numerical Digits

OVERVIEW

This project explored K-means clustering classification of numerical digits, represented by 64 features indicating a bitmap of black and white pixels from each digit’s image.

IMPLEMENTATION

This project was coded in Java 1.7.0\_09, using Eclipse on a MacBook Pro running Mavericks. Training and test data was stored in external files, read in and assimilated into respective arrays. Initial seed clusters were randomly selected from existing training instances. Using sum-squared error calculations across all training instances, clusters were then updated per their members’ features until their movement converged below a specified threshold (tested at 0.01) - averaged across all features, across all clusters - or until a cap 1000 iterations elapsed (to prevent infinite looping).

This entire process was repeated for five iterations to obtain the lowest SSE of all five. The total SSE, sum-squared separation of the resulting clusters, and the mean entropy for the training data was then calculated.

Once the clusters had been trained, they were used to classify the test data, calculating the final accuracy and confusion matrix.

RESULTS (BEST ITERATION)

Training Data:

SSE: 2472874

SSS: 59301

Mean Entropy: 0.9404487673629774

Test Data:

1797 total instances

Accuracy: 74.18%

0 1 2 3 4 5 6 7 8 9

0 176 0 0 0 2 0 0 0 0 0

1 0 62 21 1 0 0 4 0 94 0

2 1 2 149 9 0 0 0 3 13 0

3 0 0 0 165 0 2 0 6 7 3

4 0 4 0 0 151 0 0 3 8 15

5 0 0 0 22 1 156 1 0 0 2

6 1 1 0 0 1 0 176 0 2 0

7 0 0 0 0 0 0 0 140 3 36

8 0 6 1 23 0 2 1 1 129 11

9 0 3 0 144 0 3 0 0 1 29

Cluster Visualization

9

0 0 2 9 12 10 5

0 1 10 13 8 11 9

0 3 12 7 5 12 8

0 3 10 10 11 14 6

0 1 5 8 11 13 3

0 0 1 3 12 8 1

0 0 1 8 13 4 0

0 0 2 11 8 1 0

1

0 0 0 3 12 10 1

0 0 1 8 15 13 2

0 1 6 13 15 12 1

0 4 12 14 15 11 1

0 3 8 9 15 11 1

0 1 2 6 14 11 1

0 0 1 5 14 12 2

0 0 0 3 12 12 2

7

0 0 8 14 14 12 6

0 1 9 10 10 13 7

0 1 2 1 7 13 4

0 1 4 7 13 12 3

0 2 9 14 15 12 4

0 1 4 13 10 3 1

0 0 6 14 4 0 0

0 0 9 11 1 0 0

3

0 0 8 14 13 5 0

0 3 13 10 11 10 1

0 3 8 5 10 10 1

0 1 6 11 13 10 2

0 0 2 6 9 13 4

0 0 1 1 2 12 7

0 1 8 6 7 13 7

0 0 8 14 14 9 2

0

0 0 5 13 11 3 0

0 1 13 13 12 11 1

0 4 14 5 3 13 3

0 5 13 2 0 9 7

0 5 12 1 0 9 7

0 3 14 2 1 11 6

0 1 13 10 11 13 3

0 0 5 14 13 5 0

4

0 0 0 5 12 3 0

0 0 2 12 10 2 1

0 0 7 13 4 5 4

0 3 13 8 4 11 6

0 7 15 9 11 15 5

0 8 12 12 15 13 2

0 2 4 7 15 7 0

0 0 0 6 13 3 0

5

0 1 8 13 14 12 3

0 3 14 13 10 8 2

0 5 14 6 1 0 0

0 5 14 12 8 3 0

0 2 7 9 10 7 1

0 0 1 3 8 10 1

0 1 6 8 12 9 1

0 0 9 14 10 2 0

8

0 0 5 13 12 5 0

0 2 11 13 12 10 2

0 3 11 10 11 9 1

0 1 9 14 14 6 0

0 1 8 15 13 4 0

0 1 10 12 11 6 1

0 1 11 11 11 8 1

0 0 5 13 12 6 1

6

0 0 2 11 8 1 0

0 0 8 14 6 1 0

0 1 13 9 1 0 0

0 2 14 7 3 1 0

0 3 15 12 11 9 2

0 2 15 11 8 11 9

0 1 11 13 6 11 11

0 0 2 11 15 13 6

2

0 1 11 14 7 1 0

0 5 14 13 12 3 0

0 5 8 5 12 4 0

0 1 3 5 12 4 0

0 0 1 8 11 2 0

0 0 4 11 8 1 0

0 1 12 15 12 11 7

0 1 11 14 13 12 9

OBSERVATIONS

Clustering didn’t perform as well as I might have liked to see. The lowest SSE values were around 2,470,000, and mean entropy values on training data were still up around 0.95 in all runs. Further, 8s and 9s were commonly misclassified (usually as 5s and 3s, respectively); on some runs no cluster would even have this classification, leading to no instances of these classes being correctly labeled.

However, I did see one run that yielded an 80.8% accuracy and 0.839 mean entropy. I attribute this to a fortunate selection of seed cluster centers.

The visualization was interesting, and most do look somewhat like their respective digits. However, the perceived image isn’t completely clear, so it takes a little bit of imagination or a label to be able to identify it.

K SET TO 30

Training Data:

SSE: 1790104

SSS: 653524

Mean Entropy: 0.39891750286522965

Test Data:

1797 total instances

Accuracy: 90.93%

0 1 2 3 4 5 6 7 8 9

0 177 0 0 0 1 0 0 0 0 0

1 0 149 23 1 0 0 2 0 4 3

2 1 5 161 2 0 0 0 2 5 1

3 0 0 1 146 0 4 0 3 6 23

4 0 6 0 0 171 0 1 0 3 0

5 0 0 0 0 1 177 1 0 0 3

6 2 2 0 0 0 1 174 0 2 0

7 0 0 0 0 1 0 0 167 2 9

8 0 20 0 0 0 1 0 1 147 5

9 0 2 0 2 0 5 0 2 4 165

Cluster Visualization

0

0 0 7 13 6 0 0

0 1 14 14 14 6 0

0 4 14 6 9 14 2

0 5 13 1 1 12 6

0 6 12 0 0 10 7

0 4 14 1 1 12 7

0 1 15 10 11 15 3

0 0 6 14 13 6 0

8

0 0 8 13 11 6 1

0 3 14 10 9 12 3

0 3 12 7 10 12 3

0 1 9 13 13 6 1

0 0 7 15 10 1 0

0 1 12 12 10 1 0

0 1 14 10 11 3 0

0 0 9 14 10 2 0

4

0 0 0 3 13 4 0

0 0 0 8 13 2 0

0 0 2 13 8 3 2

0 0 9 12 3 9 4

0 5 15 6 7 14 4

1 12 15 13 15 15 3

1 6 9 10 15 11 1

0 0 0 4 14 5 0

9

0 0 7 14 11 3 0

0 3 15 9 12 11 1

0 6 14 3 8 14 3

0 3 13 11 12 15 4

0 0 4 7 7 14 6

0 0 0 0 1 12 7

0 1 6 6 6 13 6

0 0 7 14 14 9 1

1

0 0 0 2 11 11 1

0 0 1 7 15 14 2

0 1 6 12 15 12 1

0 4 12 13 15 11 1

0 3 9 8 15 11 1

0 1 2 4 14 11 1

0 0 0 3 14 12 2

0 0 0 2 12 13 3

1

0 0 2 13 12 2 0

0 0 7 16 15 4 0

0 2 10 15 14 3 0

0 2 10 15 14 2 0

0 1 8 15 13 2 0

0 0 7 14 13 2 0

0 0 6 14 14 5 0

0 0 2 12 13 6 1

4

0 0 2 12 8 1 1

0 0 9 14 4 5 5

0 3 14 8 2 10 6

0 7 15 9 11 15 7

0 5 13 14 16 13 4

0 1 3 9 15 5 0

0 0 1 11 11 1 0

0 0 2 13 7 0 0

9

0 0 1 7 13 12 6

0 1 10 13 9 12 9

0 3 14 9 8 14 8

0 3 12 11 12 15 6

0 0 3 4 7 14 3

0 0 0 1 10 10 1

0 0 0 5 13 5 0

0 0 1 8 9 2 0

3

0 1 8 13 15 10 2

0 4 13 10 10 15 4

0 1 3 2 10 13 2

0 0 1 8 14 7 0

0 0 1 7 13 10 2

0 0 0 1 7 13 4

0 1 5 7 10 13 3

0 1 9 14 12 5 0

2

0 1 10 15 9 1 0

0 5 15 12 14 4 0

0 4 7 3 14 4 0

0 1 1 6 14 2 0

0 0 1 12 9 0 0

0 0 8 14 3 0 0

0 2 14 14 10 10 5

0 1 10 14 14 13 7

2

0 1 12 12 1 0 0

0 6 14 15 6 0 0

0 6 9 11 9 0 0

0 2 5 11 9 0 0

0 0 3 12 7 0 0

0 0 5 11 6 2 0

0 2 12 14 14 13 8

0 1 12 14 13 13 9

5

0 1 9 13 13 9 1

0 3 14 14 13 10 2

0 1 11 12 3 1 0

0 0 6 14 4 0 0

0 0 2 11 8 1 0

0 0 2 8 12 1 0

0 1 8 12 12 1 0

0 1 11 15 7 0 0

5

0 0 8 13 14 12 4

0 2 14 13 10 8 2

0 4 14 6 1 0 0

0 5 15 14 11 3 0

0 2 8 8 13 10 1

0 0 0 0 6 13 2

0 1 5 7 12 12 1

0 0 8 15 12 4 0

6

0 0 5 12 3 0 0

0 0 12 11 1 0 0

0 2 14 5 0 0 0

0 4 14 4 3 1 0

0 6 14 12 13 11 2

0 4 16 12 8 12 9

0 1 14 12 7 13 9

0 0 5 13 15 11 2

6

0 0 0 9 11 2 0

0 0 5 15 9 1 0

0 0 11 12 1 0 0

0 1 13 8 0 0 0

0 2 14 10 8 5 1

0 1 14 14 13 14 7

0 0 9 15 8 11 12

0 0 1 8 14 14 8

9

0 0 7 14 12 3 0

0 2 12 12 12 8 1

0 2 10 9 11 11 1

0 0 4 10 13 13 3

0 0 0 2 5 12 7

0 0 1 0 1 9 11

0 1 8 6 6 12 11

0 0 6 14 15 13 5

6

0 0 1 11 10 1 0

0 0 7 15 7 1 0

0 1 12 9 0 0 0

0 2 14 9 5 2 0

0 2 15 14 13 13 4

0 1 14 9 2 7 12

0 0 9 12 4 9 13

0 0 1 10 15 14 6

0

0 0 4 14 12 3 0

0 2 14 11 10 12 1

0 5 14 3 1 12 4

0 6 12 1 0 7 8

0 6 11 0 0 6 9

0 3 13 1 0 8 8

0 1 13 8 7 14 4

0 0 4 14 15 8 0

7

0 1 9 14 15 10 3

0 1 10 10 13 13 3

0 1 2 1 11 11 1

0 1 5 9 15 10 3

0 2 10 15 15 12 5

0 1 5 14 9 4 1

0 0 7 14 3 0 0

0 1 11 10 1 0 0

8

0 0 6 13 12 4 0

0 3 13 9 11 10 1

0 4 11 7 12 8 0

0 2 10 14 12 3 0

0 1 9 13 11 5 1

0 1 11 4 6 11 4

0 1 12 5 4 12 7

0 0 6 13 14 11 3

3

0 1 10 15 12 3 0

0 5 13 9 14 8 0

0 2 3 4 13 6 0

0 0 3 12 14 5 0

0 0 2 8 13 13 3

0 0 0 0 2 13 8

0 1 9 6 7 14 7

0 1 11 15 14 9 1

2

0 1 9 14 8 1 0

0 4 15 12 14 5 0

0 4 10 2 10 8 0

0 1 3 1 11 8 0

0 0 0 2 13 5 0

0 0 1 8 13 3 0

0 0 9 15 14 11 8

0 1 10 14 13 12 11

5

0 0 8 13 14 11 4

0 2 14 10 6 6 2

0 4 14 7 6 3 1

0 4 14 14 14 12 1

0 1 3 2 3 13 5

0 0 1 0 1 12 6

0 1 7 6 8 13 3

0 0 8 14 12 5 0

4

0 0 0 5 13 2 0

0 0 1 13 10 1 2

0 0 9 13 2 5 6

0 4 15 6 3 12 7

0 9 15 10 13 15 6

0 6 11 12 15 13 2

0 1 1 5 15 5 0

0 0 0 6 13 2 0

7

0 0 5 13 14 14 10

0 0 8 10 9 13 12

0 0 1 1 2 13 8

0 0 2 5 10 14 5

0 1 7 13 15 13 4

0 1 3 11 13 4 0

0 0 3 13 7 0 0

0 0 6 13 3 0 0

1

0 0 4 12 12 5 0

0 1 8 15 16 11 1

0 1 8 15 16 12 1

0 0 8 15 16 11 1

0 1 7 15 16 10 0

0 0 7 15 16 10 1

0 0 8 15 16 11 1

0 0 4 12 12 7 1

0

0 0 2 13 12 2 0

0 1 10 15 14 10 0

0 3 14 9 3 14 2

0 4 14 6 0 10 6

0 3 14 4 0 9 8

0 1 15 5 1 12 8

0 0 10 13 11 15 4

0 0 2 12 14 7 0

0

0 0 5 13 12 4 0

0 1 13 14 12 13 1

0 3 15 4 2 12 4

0 5 13 1 0 9 7

0 5 12 0 0 11 6

0 4 13 2 3 14 3

0 1 14 12 14 11 0

0 0 5 14 11 2 0

8

0 0 4 12 13 7 1

0 2 14 10 9 13 3

0 4 13 5 5 13 3

0 1 10 13 14 9 1

0 0 7 15 14 6 0

0 1 11 9 8 11 2

0 1 12 9 7 14 3

0 0 5 13 14 9 1

5

0 1 9 14 14 12 3

0 5 15 14 11 9 3

0 8 14 4 1 0 0

0 6 15 8 3 0 0

0 2 10 12 9 1 0

0 0 2 7 12 4 0

0 0 6 11 12 4 0

0 1 10 14 7 1 0

Increasing the cluster count to 30 greatly increased the classification accuracy. More clusters allow for more precise classification, especially if it means that all classifications have a higher probability of being represented (with only 10 clusters, if any classes were represented by more than one cluster, another would be left unrepresented entirely, knocking out entire class that could be classified correctly).

CONCLUSION

K-means clustering with only 10 clusters was effective enough to address a majority of the cases, still achieving around a 75% accuracy. Classification was efficient, finishing much faster than Naïve Bayes. However, the latter was more accurate, correctly classifying instances in almost 90% of all cases. This, in combination with the fact that some classifications are completely dropped with the 10-cluster K-means (i.e.: all classes aren’t always completely represented by the clusters), would make Naïve Bayes the preferred approach, despite the longer execution time.

On the other hand, increasing the number of clusters to 30 brought K-means back into the realm of feasible classification techniques. Overall accuracy was over 90%, and the mean entropy dropped significantly to under 0.4. Cluster visualizations don’t really seem much clearer than with 10 clusters. In the end, this approach provided the same level of accuracy with less time than Naïve Bayes, proving the better implementation.