Artificial Intelligence HW 3

Problem 1
a. See included Python implementation
b. The Initial Clusters image shows the initial clusters (in the center of which lies the initial randomly chosen centroids). The Final Clusters image shows the clusters found by the k-means algorithm after 100 iterations. The results shown make sense, as the k-means algorithm will find the "local minimum" for its final centroid. Thus, the final clusters found will depend heavily on which initial centroids are selected.
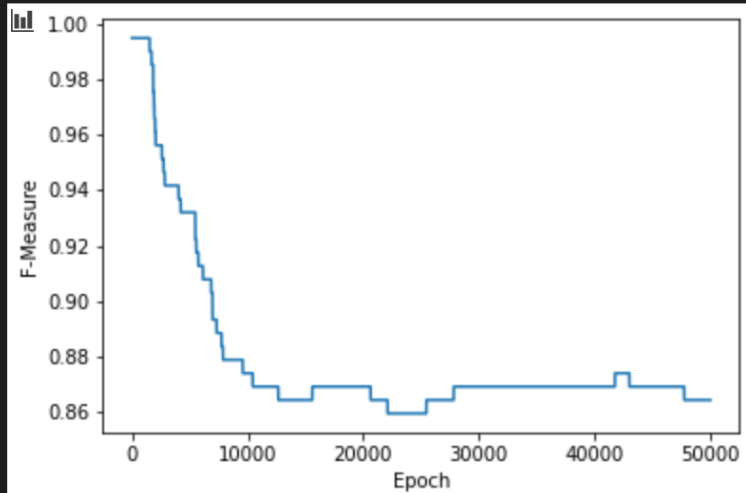


Initial Clusters:



Final Clusters:
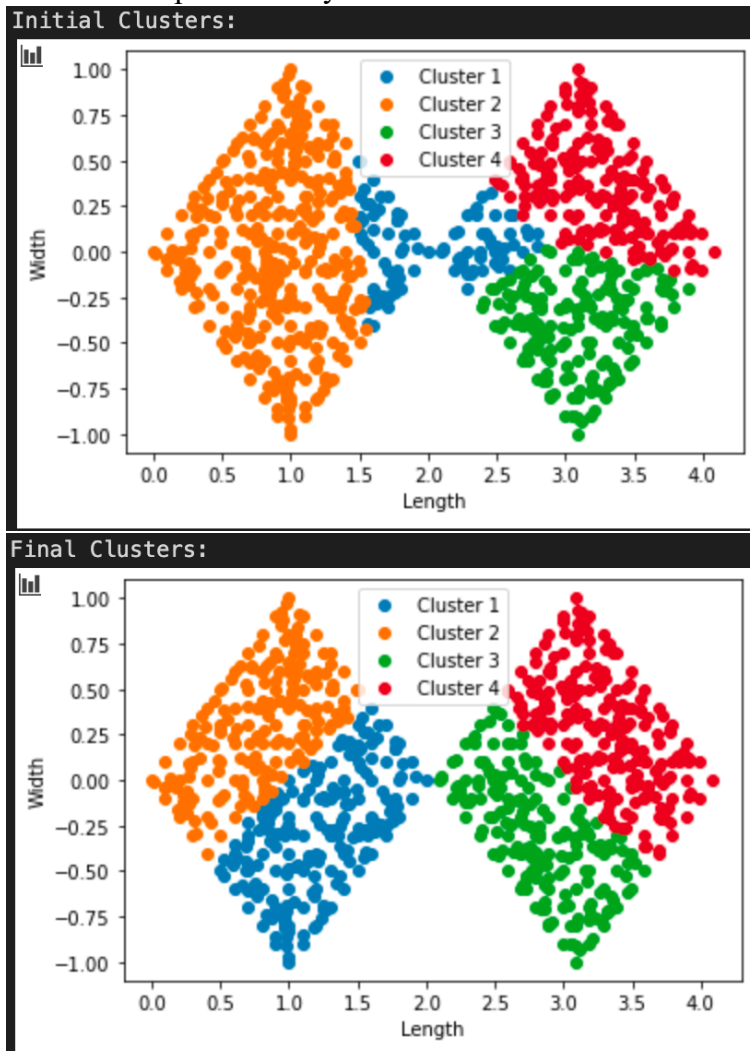
Problem 2
a. See included Python implementation

b.

```
X_train : (320, 24)
y_train : (320,)
X_test : (80, 24)
y_test : (80,)
```
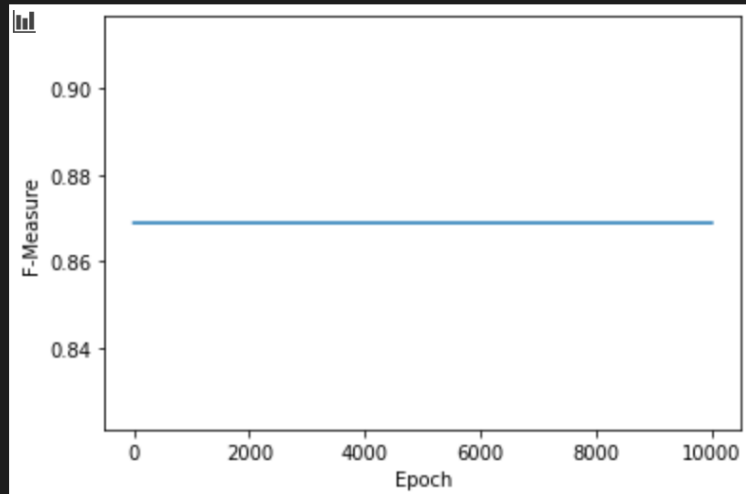


```
LR Accuracy on Training Data:  0.821875
LR Accuracy on Test Data:  0.75
```
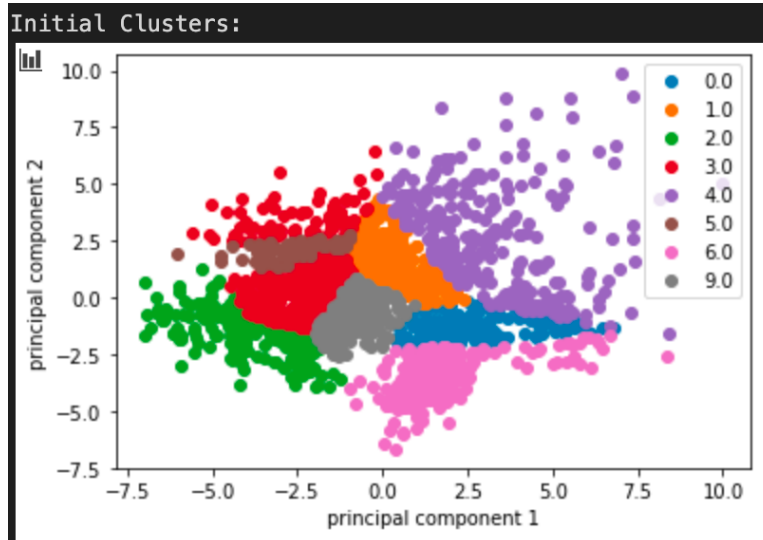
c.

```
Data mean should be near 0: 1.258252761241844e-17
Data standard deviation should be 1: 1.0
------
X_train : (320, 24)
y_train : (320,)
X_test : (80, 24)
y_test : (80,)
```
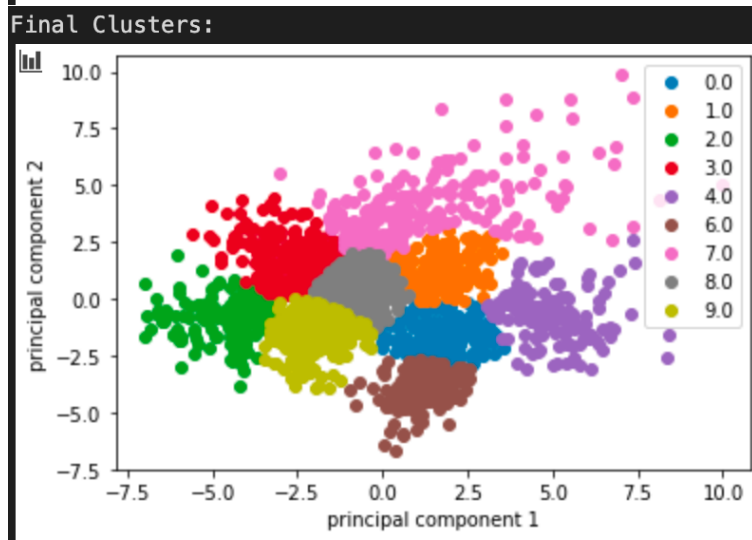


```
LR Accuracy on Training Data:  0.90625
LR Accuracy on Test Data:  0.9
```

Problem 3



Initial Clusters:

a.

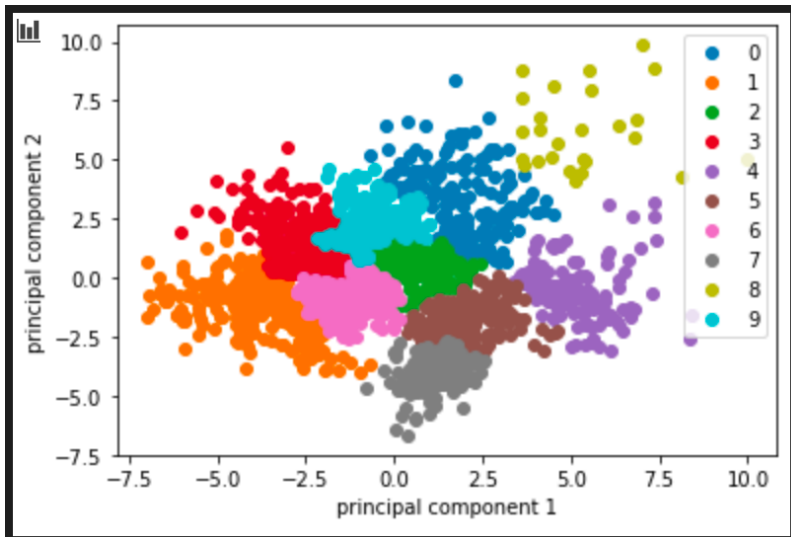Final Clusters:



```
Actual vs. Predicted Data Array % Similarity:
90.0%
Confusion Matrix:
[[124   6   0   0   5   0  40   0   1   2]
 [  6  74   0   0   0   0   2  10  63  27]
 [  1   8 106  16   0   0   0   1  10  35]
 [  4   0  38  76   0   0   0  15  10  40]
 [ 12   8   0   2 150   0   0   8   1   0]
 [  8  15   2  60   1   0   0  13  60  23]
 [ 21   1   0   0   0   0 159   0   0   0]
 [  0  14   0  12   0   0   0 146   7   0]
 [  9  27   2  25   0   0   0  10  75  26]
 [  2   8   7  15   0   0   0  29  61  58]]
Fowlkes-Mallows Score:
0.41134857985642637
```

b.



```
Confusion Matrix:
[[   0    1   16    0    1  123    2   35    0    0]
 [  54   22   74    0    0    0   16    1    0   15]
 [   4  128    6   16    0    0   21    0    0    2]
 [   4   53    1   76    0    2   32    0    0   15]
 [   5    0    5    2  135   29    0    0    4    1]
 [   9    7   27   67    1    5   36    0    0   30]
 [   0    0    0    0    0   25    0  156    0    0]
 [  80    0    8    8    0    0    0    0   15   68]
 [   9    6   51   25    0    0   52    0    0   31]
 [  19   19   15   27    0    0   78    0    7   15]]
Fowlkes-Mallows Score:
0.38253228686622304
```
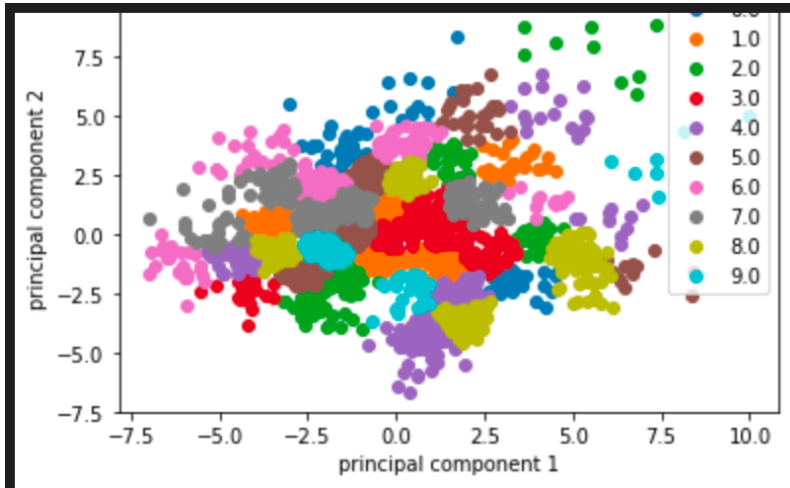
c.

```
Confusion Matrix:
[[ 11  39   3  38  52   0   0   0  14  21]
 [  0  47  34  51   1   1   1  39   7   1]
 [  0  19   4  22  24  19  31  27  25   6]
 [ 11  30   6   1  11  16  25  47  17  19]
 [ 11   1  37  19  10  13  12   6  65   7]
 [  8  27   9  27   0  23  26  47   3  12]
 [  0   5   0   2 123   0   0   0  49   2]
 [ 22   9  19   5  10  44  42   7  21   0]
 [  0  41   4  43   1  24  12  29  10  10]
 [  1  42  19  18   6  31  11  24  10  18]]
Fowlkes-Mallows Score:
0.19317745499905928
```

Note: the above values are bad because of the nature of Affinity Propagation. It fails at classifying the values by clustering things correctly as it does not limit itself to 10 clusters. The above images are the results after normalizing the labels provided by the algorithm.

Problem 4:

```
X_train : (320, 24)
y_train : (320,)
X_test : (80, 24)
y_test : (80,)
F-Measure: 0.9885057471264368
```

a.

```
X_train : (320, 24)
y_train : (320,)
X_test : (80, 24)
y_test : (80,)
F-Measure: 0.9885057471264368
```

b.

```
X_train : (320, 24)
y_train : (320,)
X_test : (80, 24)
y_test : (80,)
F-Measure: 1.0
```

c.