Report for Assignment 1 Part 1
“K-nearest method”

1. Report the labels and accuracy when k=1

Instance 1: 3

Instance 2: 3

Instance 3: 3

Instance 4: 1

Instance 5: 1

Instance 6: 1

Instance 7: 1

Instance 8: 2

Instance 9: 1

Instance 10: 2

Instance 11: 2

Instance 12: 3

Instance 13: 3

Instance 14: 3

Instance 15: 1

Instance 16: 2

Instance 17: 3

Instance 18: 3

Instance 19: 1

Instance 20: 1

Instance 21: 3

Instance 22: 2

Instance 23: 2

Instance 24: 3

Instance 25: 2

Instance 26: 3

Instance 27: 2

Instance 28: 3

Instance 29: 2

Instance 30: 1

Instance 31: 2

Instance 32: 1

Instance 33: 2

Instance 34: 1

Instance 35: 2

Instance 36: 2

Instance 37: 2

Instance 38: 2

Instance 39: 2

Instance 40: 1

Instance 41: 2

Instance 42: 2

Instance 43: 3

Instance 44: 1

Instance 45: 2

Instance 46: 1

Instance 47: 3

Instance 48: 2

Instance 49: 2

Instance 50: 1

Instance 51: 3

Instance 52: 1

Instance 53: 1

Instance 54: 3

Instance 55: 3

Instance 56: 1

Instance 57: 1

Instance 58: 3

Instance 59: 1

Instance 60: 3

Instance 61: 3

Instance 62: 1

Instance 63: 2

Instance 64: 3

Instance 65: 2

Instance 66: 3

Instance 67: 3

Instance 68: 1

Instance 69: 1

Instance 70: 2

Instance 71: 1

Instance 72: 3

Instance 73: 2

Instance 74: 2

Instance 75: 1

Instance 76: 1

Instance 77: 1

Instance 78: 3

Instance 79: 1

Instance 80: 1

Instance 81: 2

Instance 82: 2

Instance 83: 3

Instance 84: 1

Instance 85: 2

Instance 86: 1

Instance 87: 1

Instance 88: 2

Instance 89: 1

Accuracy: 94%

2. Report the labels with k =3 and compare:

The result appeared to be 2% more accurate than in 1-nearest neighbour method. If k=1, result is assigned to the first nearest neighbour and it could lead to mistakes as rely only on 1 point. K=3 makes it more stable as result depends on 3 points which makes it more reliable.

Instance 1: 3
Instance 2: 3
Instance 3: 3
Instance 4: 1
Instance 5: 1
Instance 6: 1
Instance 7: 2
Instance 8: 2
Instance 9: 1
Instance 10: 2
Instance 11: 3
Instance 12: 3
Instance 13: 3
Instance 14: 3
Instance 15: 1
Instance 16: 2
Instance 17: 3
Instance 18: 3
Instance 19: 1
Instance 20: 1
Instance 21: 3
Instance 22: 2
Instance 23: 2
Instance 24: 3
Instance 25: 2
Instance 26: 3
Instance 27: 2
Instance 28: 3
Instance 29: 2
Instance 30: 1
Instance 31: 2
Instance 32: 1
Instance 33: 2
Instance 34: 1
Instance 35: 2
Instance 36: 2
Instance 37: 2
Instance 38: 2
Instance 39: 2
Instance 40: 1
Instance 41: 2
Instance 42: 2
Instance 43: 3
Instance 44: 1
Instance 45: 2
Instance 46: 1
Instance 47: 3
Instance 48: 2
Instance 49: 2
Instance 50: 1
Instance 51: 3
Instance 52: 1
Instance 53: 1
Instance 54: 3
Instance 55: 3
Instance 56: 1
Instance 57: 1
Instance 58: 3
Instance 59: 1
Instance 60: 3
Instance 61: 3
Instance 62: 1
Instance 63: 2
Instance 64: 3
Instance 65: 2
Instance 66: 3
Instance 67: 3
Instance 68: 1
Instance 69: 1
Instance 70: 2
Instance 71: 2
Instance 72: 3
Instance 73: 2
Instance 74: 2
Instance 75: 1
Instance 76: 1
Instance 77: 1
Instance 78: 3
Instance 79: 1
Instance 80: 1
Instance 81: 2
Instance 82: 2
Instance 83: 3
Instance 84: 1
Instance 85: 2
Instance 86: 1
Instance 87: 1
Instance 88: 2
Instance 89: 1
Accuracy: 96%

3. Discuss the main advantages and disadvantages of k-Nearest Neighbour method.

Advantages	Disadvantages
Construct hypotheses directly from the training instances themselves	Cannot be trusted for high dimensional data
Can be very simply implemented by only adding k and distance measuring method	Not very quick with huge data sets
New sets could be added easily	

4. Assuming that you are asked to apply the k-fold cross validation method for the above problem with
k=5, what would you do? State the major steps.

1. Divide the data set into 5 equal subsets
2. Treat each of them as test set and other 4 as training set.
3. Train classifier on training set and test on test set.
4. Repeat the process 5 times.
5. Combine or average 5 results to single estimation.

5. In the above problem, if there were actually no class labels available. Which method would you use to group the examples in the data set? State the major steps.

With not labelling data k-mean clustering method could be used.

1. Initialise k initial “means” randomly from the data set
2. Create k clusters by assigning every instance to the nearest cluster: based on the nearest mean according to the distance measure
3. Replace the old means with the centroid(mean) of each cluster
4. Repeat the above two steps until convergence (no change in each cluster centroid).

6. Clustering method

6.1 Training set

The programme produces the following report for 3 clusters:

`python assign-1-1-2.py wine-training 3` or just `python assign-1-1-2.py wine-training`

```
The convergence of centroids hapenned on 5 iteration
Final centroids coordinates:
[[13.860740740740741, 1.9544444444444442, 2.4492592592592595, 16.503703703703703,
108.07407407407408, 2.9318518518518513, 3.0459259259259257, 0.2948148148148148,
1.9025925925925928, 5.888148148148149, 1.0481481481481483, 3.08, 1140.7407407407406],
[12.254722222222222, 1.9416666666666669, 2.1991666666666663, 19.544444444444444, 97.25,
2.2744444444444447, 2.088055555555555, 0.335, 1.7469444444444446, 2.9308333333333336,
1.0508333333333333, 2.8466666666666662, 525.6388888888889], [13.075769230769229,
2.931538461538462, 2.43, 21.315384615384616, 100.07692307692308, 1.6965384615384618,
0.8096153846153846, 0.4646153846153846, 1.195, 7.523076884615386, 0.697923076923077,
1.7523076923076923, 654.4230769230769]]
The size of the final 3 clusters is [27, 36, 26]
The rate of right guesses for class 1 is 93.10344827586206%
The rate of right guesses for class 2 is 94.44444444444444%
The rate of right guesses for class 3 is 100.0%
```

6.2. Test set

Tested on test set.
python assign-1-1-2.py wine-test 3 or just python assign-1-1-2.py wine-test

```
The convergence of centroids happened on 3 iteration
Final centroids coordinates:
[[13.159999999999998, 3.617692307692308, 2.397307692307692, 21.134615384615383,
97.42307692307692, 1.6488461538461539, 0.8357692307692309, 0.43615384615384617,
1.1088461538461538, 6.785384615384617, 0.6942307692307692, 1.6457692307692307,
593.3461538461538], [13.554117647058826, 1.9020588235294122, 2.41, 17.694117647058825,
103.97058823529412, 2.8329411764705883, 2.9520588235294127, 0.2705882352941763,
1.9102941176470585, 5.164999999999999, 1.0844117647058822, 3.221470588235293, 1075.0],
[12.266551724137932, 2.008620689655172, 2.3617241379310348, 21.227586206896554,
91.89655172413794, 2.2134482758620693, 2.091379310344828, 0.4058620689655172,
1.5196551724137934, 3.0420689655172413, 1.0768965517241138, 2.8055172413793104,
490.7586206896552]]
The size of the final 3 clusters is [26, 34, 29]
The rate of right guesses for class 1 is 0.0%
The rate of right guesses for class 2 is 0.0%
The rate of right guesses for class 3 is 11.428571428571429%
```

The number or correct guesses was extremely low.
However, real classes and clusters look similar.

```
!!!!!!!!!!!!!!!!!!!!
{0: [0, 1, 2, 11, 12, 13, 16, 17, 20, 23, 25, 27, 42, 46, 50, 53, 54, 57, 59, 60, 63, 65, 66, 71, 77, 82], 1: [3,
4, 5, 7, 8, 14, 18, 19, 29, 31, 33, 39, 43, 45, 48, 49, 51, 52, 55, 56, 58, 61, 64, 67, 68, 74, 75, 76, 78, 79,
83, 85, 86, 88], 2: [6, 9, 10, 15, 21, 22, 24, 26, 28, 30, 32, 34, 35, 36, 37, 38, 40, 41, 44, 47, 62, 69, 70,
72, 73, 80, 81, 84, 87]}
{2: [0, 1, 2, 12, 13, 16, 17, 20, 23, 27, 42, 46, 50, 53, 54, 57, 59, 60, 63, 65, 66, 71, 77, 82], 0: [3, 4, 5, 8,
14, 18, 19, 29, 31, 33, 39, 43, 45, 49, 51, 52, 55, 56, 58, 67, 68, 74, 75, 76, 78, 79, 83, 85, 86, 88], 1: [6,
7, 9, 10, 11, 15, 21, 22, 24, 25, 26, 28, 30, 32, 34, 35, 36, 37, 38, 40, 41, 44, 47, 48, 61, 62, 64, 69, 70,
72, 73, 80, 81, 84, 87]}
!!!!!!!!!!!!!!!!!!!!!!!
```

After swapping indexes around to compare with similar clusters the result was as high as for the training set

```
The rate of right guesses for class 1 is 100.0%
The rate of right guesses for class 2 is 100.0%
The rate of right guesses for class 3 is 82.85714285714286%
```

6.3. 5 clusters

```
python assign-1-1-2.py wine-test 5
```

Could be any number of clusters.

```
The convergence of centroids happened on 10 iteration
Final centroids coordinates:
[[13.159999999999998, 3.617692307692308, 2.397307692307692, 21.134615384615383,
97.42307692307692, 1.6488461538461539, 0.8357692307692309, 0.43615384615384617,
1.1088461538461538, 6.785384615384617, 0.6942307692307692, 1.6457692307692307,
593.3461538461538], [13.285555555555554, 1.7461111111111111, 2.2027777777777775,
17.044444444444444, 95.22222222222223, 2.4916666666666667, 2.5799999999999996,
0.28166666666666666, 1.6044444444444443, 4.291111111111111, 1.1016666666666666,
3.143888888888889, 908.7777777777778], [13.934375, 1.978125, 2.510625, 17.3125, 109.625, 2.95875,
3.20375, 0.27375, 2.1181249999999996, 6.150625000000001, 1.0950000000000002, 3.19125, 1208.5625],
[12.224545454545455, 2.3654545454545453, 2.4736363636363636, 21.418181818181818,
100.9090909090909, 2.8827272727272724, 2.789090909090909, 0.2654545454545455,
1.9336363636363638, 3.4709090909090907, 1.0127272727272727, 3.2818181818181817,
563.8181818181819], [12.222777777777777, 1.8788888888888886, 2.411111111111111, 22.1,
90.11111111111111, 2.033888888888889, 1.8133333333333337, 0.47777777777777775,
1.3877777777777776, 2.7777777777777777, 1.0894444444444447, 2.6188888888888884,
493.6111111111111]]
The size of the final 5 clusters is [26, 18, 16, 11, 18]
!!!!!!!!!!!!!!!!!!!!
```

Clusters

```
{0: [0, 1, 2, 11, 12, 13, 16, 17, 20, 23, 25, 27, 42, 46, 50, 53, 54, 57, 59, 60, 63, 65, 66, 71, 77, 82], 1: [3, 5, 7, 14,
19, 21, 34, 40, 45, 51, 58, 67, 69, 74, 78, 79, 83, 86], 2: [4, 8, 18, 29, 31, 33, 39, 43, 49, 52, 55, 68, 75, 76, 85,
88], 3: [6, 28, 35, 48, 56, 61, 62, 64, 70, 84, 87], 4: [9, 10, 15, 22, 24, 26, 30, 32, 36, 37, 38, 41, 44, 47, 72, 73, 80,
81]}
```

Classes

```
{2: [0, 1, 2, 12, 13, 16, 17, 20, 23, 27, 42, 46, 50, 53, 54, 57, 59, 60, 63, 65, 66, 71, 77, 82], 0: [3, 4, 5, 8, 14, 18,
19, 29, 31, 33, 39, 43, 45, 49, 51, 52, 55, 56, 58, 67, 68, 74, 75, 76, 78, 79, 83, 85, 86, 88], 1: [6, 7, 9, 10, 11, 15,
21, 22, 24, 25, 26, 28, 30, 32, 34, 35, 36, 37, 38, 40, 41, 44, 47, 48, 61, 62, 64, 69, 70, 72, 73, 80, 81, 84, 87]}
!!!!!!!!!!!!!!!!!!!!!!!
```

Value of class == index in dictionary +1

```
The rate of right guesses for class 1 is 100.0%
The rate of right guesses for class 2 is 43.333333333333336%
The rate of right guesses for class 3 is 0.0%
```

The convergence also happened however it does not make a lot of sense during real class identification. As there is only 3 classes.