

Accuracy: 95.1%

Confusion matrix:

```
[[116  0  0  0  0  0  0  0  0  0]
 [  0 94  0  0  0  0  0  1  0  0]
 [  0  1 93  0  0  0  0  1  1  0]
 [  0  1  1 87  0  2  1  3  1  0]
 [  0  1  0  0 85  0  0  0  0  7]
 [  0  1  0  2  0 113  1  0  0  1]
 [  1  0  0  0  0  2 81  0  0  0]
 [  0  1  0  0  1  1  0 93  0  1]
 [  0  3  0  3  1  1  2  0 102  0]
 [  1  1  1  0  1  0  1  1  0 87]]
```

Per-class precision and recall:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	116
1	0.91	0.99	0.95	95
2	0.98	0.97	0.97	96
3	0.95	0.91	0.93	96
4	0.97	0.91	0.94	93
5	0.95	0.96	0.95	118
6	0.94	0.96	0.95	84
7	0.94	0.96	0.95	97
8	0.98	0.91	0.94	112
9	0.91	0.94	0.92	93
micro avg	0.95	0.95	0.95	1000
macro avg	0.95	0.95	0.95	1000
weighted avg	0.95	0.95	0.95	1000

Extra Task:

I have tried different k values, range from 1 to 14:

```
1 neighbours selected, the accuracy of the model is 0.953
2 neighbours selected, the accuracy of the model is 0.94
3 neighbours selected, the accuracy of the model is 0.955
4 neighbours selected, the accuracy of the model is 0.953
5 neighbours selected, the accuracy of the model is 0.951
6 neighbours selected, the accuracy of the model is 0.952
7 neighbours selected, the accuracy of the model is 0.95
8 neighbours selected, the accuracy of the model is 0.951
9 neighbours selected, the accuracy of the model is 0.947
10 neighbours selected, the accuracy of the model is 0.946
11 neighbours selected, the accuracy of the model is 0.947
12 neighbours selected, the accuracy of the model is 0.945
13 neighbours selected, the accuracy of the model is 0.94
14 neighbours selected, the accuracy of the model is 0.939
```

And I realize, that when $k = 3$, the model has the highest accuracy, the accuracy drops with larger k values selected, as it is likely to be affected by the outliers.