

CPU: AMD R7 Radeon

Graphic Cards: NVIDIA GeForce GTX 1060

Ram: 8GB

Random Forest:

By using `sklearn.ensemble.RandomForestClassifier`, I fit the data in the model and calculate the accuracy of training and testing dataset of fitted model comparing to true value. The performance is very good for both training and testing dataset.

Accuracy:

Training Dataset: 0.993

Testing Dataset: 0.82

Running time: 28.862 second

SVM:

By using `sklearn.svm.SVC`, I fit the data in the model using kernel Radial Basic Function (RBF) and calculate the accuracy of training and testing dataset of fitted model comparing to true value.

Before I fit the data, I carry out 5-fold Cross-Validation to decide the value of Gamma to be used in SVM. The Gamma generates highest accuracy will be selected for fitting testing dataset. However, no matter how I change Gamma (or even C, the penalty parameter of misclassified) no change of accuracy can be observed. I suspect that kernel RBF is the reason for unchangeable low accuracy score due to overfitting as using kernel 'linear' have good prediction accuracy.

Accuracy:

Training Dataset: 1

Testing Dataset: 0.105

As training dataset has accuracy 1, I can reach the conclusion that the model is overfitted by training data.

Running time: 2738.507 second

Neuron Network:

By using `sklearn.neural_network.MLPClassifier`, I fit the data in the model in and calculate the accuracy of training and testing dataset of fitted model comparing to true value.

Before I fit the data, I carry out 5-fold Cross-Validation to decide the value of `hidden_layer_size` (H) to be used in `MLPClassifier`. The H generates highest accuracy will be selected for fitting testing dataset. When H equals 50, the mean accuracy generated among the 5 cross validation sets are highest. Thus, H = 50 is used when I calculate the accuracy of testing dataset.

Accuracy:

Training Dataset: 0.7994

Testing Dataset: 0.766

As training dataset has accuracy 1, I can reach the conclusion that the model is overfitted by training data.

Running time: 155.942 second

Confusion matrix and classification report

Random Forest:

Train:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	942
1	1.00	1.00	1.00	1027
2	0.98	1.00	0.99	1016
3	0.99	1.00	0.99	1019
4	0.99	0.99	0.99	974
5	0.99	1.00	1.00	989
6	1.00	0.96	0.98	1021
7	1.00	1.00	1.00	1022
8	1.00	1.00	1.00	990
9	1.00	0.99	1.00	1000
avg / total	0.99	0.99	0.99	10000

[[940	0	0	1	1	0	0	0	0	0]
[0	1027	0	0	0	0	0	0	0	0]
[0	0	1015	0	1	0	0	0	0	0]
[2	0	0	1017	0	0	0	0	0	0]
[0	0	8	3	961	0	2	0	0	0]
[0	0	0	1	0	988	0	0	0	0]
[13	0	10	3	10	0	985	0	0	0]
[0	0	0	0	0	3	0	1019	0	0]
[0	0	0	2	0	0	0	1	987	0]
[0	0	0	0	0	2	0	4	1	993]]

Test:

	precision	recall	f1-score	support
0	0.79	0.83	0.81	107
1	0.98	0.95	0.97	105
2	0.65	0.78	0.71	111
3	0.81	0.84	0.83	93
4	0.76	0.70	0.73	115
5	0.91	0.86	0.89	87
6	0.60	0.52	0.55	97
7	0.87	0.89	0.88	95
8	0.96	0.91	0.93	95
9	0.92	0.95	0.93	95
avg / total	0.82	0.82	0.82	1000

```
[[ 89  0  4  2  0  0 12  0  0  0]
 [  0 100  0  5  0  0  0  0  0  0]
 [  2  0 87  1 10  0 10  0  1  0]
 [  5  2  3 78  3  0  2  0  0  0]
 [  1  0 22  5 80  0  6  0  1  0]
 [  0  0  0  0  0 75  0 10  1  1]
 [ 13  0 17  5 12  0 50  0  0  0]
 [  0  0  0  0  0  3  0 85  0  7]
 [  2  0  0  0  0  2  4  1 86  0]
 [  0  0  0  0  0  2  0  2  1 90]]
```

SVM:

Train:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	942
1	1.00	1.00	1.00	1027
2	1.00	1.00	1.00	1016
3	1.00	1.00	1.00	1019
4	1.00	1.00	1.00	974
5	1.00	1.00	1.00	989
6	1.00	1.00	1.00	1021
7	1.00	1.00	1.00	1022
8	1.00	1.00	1.00	990
9	1.00	1.00	1.00	1000
avg / total	1.00	1.00	1.00	10000

```
[[ 942  0  0  0  0  0  0  0  0  0]
 [  0 1027  0  0  0  0  0  0  0  0]
 [  0  0 1016  0  0  0  0  0  0  0]
 [  0  0  0 1019  0  0  0  0  0  0]
 [  0  0  0  0  974  0  0  0  0  0]
 [  0  0  0  0  0  989  0  0  0  0]
 [  0  0  0  0  0  0 1021  0  0  0]
 [  0  0  0  0  0  0  0 1022  0  0]
 [  0  0  0  0  0  0  0  0  990  0]
 [  0  0  0  0  0  0  0  0  0 1000]]
```

Test:

	precision	recall	f1-score	support
0	0.00	0.00	0.00	107
1	0.10	1.00	0.19	105
2	0.00	0.00	0.00	111
3	0.00	0.00	0.00	93
4	0.00	0.00	0.00	115
5	0.00	0.00	0.00	87
6	0.00	0.00	0.00	97
7	0.00	0.00	0.00	95
8	0.00	0.00	0.00	95
9	0.00	0.00	0.00	95
avg / total	0.01	0.10	0.02	1000

```
[[ 0 107 0 0 0 0 0 0 0 0]
 [ 0 105 0 0 0 0 0 0 0 0]
 [ 0 111 0 0 0 0 0 0 0 0]
 [ 0 93 0 0 0 0 0 0 0 0]
 [ 0 115 0 0 0 0 0 0 0 0]
 [ 0 87 0 0 0 0 0 0 0 0]
 [ 0 97 0 0 0 0 0 0 0 0]
 [ 0 95 0 0 0 0 0 0 0 0]
 [ 0 95 0 0 0 0 0 0 0 0]
 [ 0 95 0 0 0 0 0 0 0 0]]
```

Neuron Network:

Train:

	precision	recall	f1-score	support
0	0.82	0.76	0.79	942
1	1.00	0.94	0.97	1027
2	0.77	0.66	0.71	1016
3	0.63	0.90	0.74	1019
4	0.57	0.80	0.67	974
5	0.99	0.88	0.93	989
6	0.52	0.28	0.37	1021
7	0.92	0.85	0.89	1022
8	1.00	0.92	0.96	990
9	0.86	0.99	0.92	1000
avg / total	0.81	0.80	0.79	10000

[[715 1 11 173 3 0 39 0 0 0]

[0 969 0 58 0 0 0 0 0 0]

[3 0 672 48 218 0 75 0 0 0]

[1 3 18 921 16 0 60 0 0 0]

[1 0 74 32 783 0 84 0 0 0]

[0 0 2 7 0 873 0 61 1 45]

[153 0 93 135 349 0 290 1 0 0]

[0 0 1 19 0 8 0 872 2 120]

[1 0 1 60 2 0 13 4 908 1]

[0 0 0 4 0 0 0 5 0 991]]

Test:

	precision	recall	f1-score	support
0	0.79	0.74	0.76	107
1	0.95	0.93	0.94	105
2	0.79	0.67	0.72	111
3	0.57	0.85	0.68	93
4	0.60	0.77	0.67	115
5	0.95	0.79	0.86	87
6	0.52	0.31	0.39	97
7	0.87	0.84	0.86	95
8	0.92	0.84	0.88	95
9	0.82	0.94	0.88	95
avg / total	0.77	0.77	0.76	1000

```
[[79 2 2 18 0 0 4 0 2 0]
 [ 0 98 0 7 0 0 0 0 0 0]
 [ 3 0 74 9 19 0 5 0 1 0]
 [ 3 2 1 79 1 0 7 0 0 0]
 [ 0 0 10 6 88 0 11 0 0 0]
 [ 0 0 1 2 0 69 0 6 2 7]
 [13 0 6 9 38 0 30 0 1 0]
 [ 0 0 0 0 0 2 0 80 1 12]
 [ 2 1 0 9 0 1 1 1 80 0]
 [ 0 0 0 0 0 1 0 5 0 89]]
```