



BLG527E Machine Learning Homework 3

Anıl Öztürk

504181504

December 19, 2018

Question	Q1	Q2	Q3	Q4	TOTAL
MaxGrade	1	1	2	1	5
ExpectedGrade	1	1	2	1	5

We need to do basic processes with our dataset, we need to classify its test values with different classification (and estimation) methods and measure the accuracies etc. Let's see the detailed request list;

- the best test equal weighted accuracy for the test set and the corresponding training set equal weighted accuracy
- the train and test confusion matrices for the case in first situation
- precision for train and test sets for the case in first situation
- the name of the library, tool and the how to call your solution

A sample confusion matrix for K=3 classes and the measures above are given as:

Actual	Predicted C1	Predicted C2	Predicted C3	Number of Instances in Each Class	Accuracy for Each Class
C1	N11	N12	N13	N1=N11+N12+N13	Acc(1)=N11/N1
C2	N21	N22	N23	N2=N21+N22+N23	Acc(2)=N22/N2
C3	N31	N32	N33	N3=N31+N32+N33	Acc(3)=N33/N3
#Instances Predicted	M1=N11+N21+N31	M2=N12+N22+N32	M3=N13+N23+N33		

The necessary formulas are below:

$$Acc_{ew} = \frac{1}{K} \sum_{i=1}^K Acc(i) \quad (0.1)$$

$$Precision = Prec(i) = \frac{N_{ii}}{M_i} \quad (0.2)$$

1 K-NN CLASSIFICATION

1.1 ACCURACIES

$$Acc_{training} = 1 \quad (1.1)$$

$$Acc_{test} = 0.98 \quad (1.2)$$

1.2 CONFUSION MATRICES (TRAINING AND TEST SET, RESPECTIVELY)

True Class	Predicted Class										Total
	0	1	2	3	4	5	6	7	8	9	
0	376	0	0	0	0	0	0	0	0	0	376
1	0	389	0	0	0	0	0	0	0	0	389
2	0	0	380	0	0	0	0	0	0	0	380
3	0	0	0	389	0	0	0	0	0	0	389
4	0	0	0	0	387	0	0	0	0	0	387
5	0	0	0	0	0	376	0	0	0	0	376
6	0	0	0	0	0	0	377	0	0	0	377
7	0	0	0	0	0	0	0	387	0	0	387
8	0	0	0	0	0	0	0	0	380	0	380
9	0	0	0	0	0	0	0	0	0	382	382
Total	376	389	380	389	387	376	377	387	380	382	3823

True Class	Predicted Class										Total
	0	1	2	3	4	5	6	7	8	9	
0	178	0	0	0	0	0	0	0	0	0	178
1	0	181	0	0	0	0	0	0	1	0	182
2	0	2	175	0	0	0	0	0	0	0	177
3	0	0	0	179	0	0	0	2	0	2	183
4	0	2	0	0	178	0	0	0	1	0	181
5	0	0	0	0	1	179	0	0	0	2	182
6	0	0	0	0	0	0	181	0	0	0	181
7	0	0	0	0	0	0	0	177	0	2	179
8	0	8	0	1	0	0	0	0	164	1	174
9	0	0	0	3	3	2	0	0	3	169	180
Total	178	193	175	183	182	181	181	179	169	176	1797

1.3 PRECISIONS

$$Prec_{training} = 1 \quad (1.3)$$

$$Prec_{training}(byclass) = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0] \quad (1.4)$$

$$Prec_{test} = 0.98 \quad (1.5)$$

$$Prec_{test}(byclass) = [1.0, 0.937, 1.0, 0.978, 0.978, 0.988, 1.0, 0.988, 0.970, 0.960] \quad (1.6)$$

1.4 EXPLANATION OF THE CODE

The assignment has been written in **Python 3.6.2** with **Sublime Text 3.1.1** IDE. In order to run the assignment code, the following Python libraries are required;

- pandas
- numpy
- warnings
- sklearn

Simply run **question1.py**. You will see precisions, accuracies and confusion matrixes as a result.

Tuning algorithm used to get best accuracy. As a result, following hyper-parameters selected for the classifier to get best accuracy result:

- k = 1

2 DECISION TREE CLASSIFICATION

2.1 ACCURACIES

$$Acc_{training} = 0.989 \quad (2.1)$$

$$Acc_{test} = 0.881 \quad (2.2)$$

2.2 CONFUSION MATRICES (TRAINING AND TEST SET, RESPECTIVELY)

	Predicted Class										
True Class	0	1	2	3	4	5	6	7	8	9	Total
0	373	0	0	0	2	1	0	0	0	0	376
1	0	388	0	1	0	0	0	0	0	0	389
2	0	0	375	1	1	0	0	0	1	2	380
3	0	2	0	386	0	0	0	0	0	1	389
4	0	1	0	1	384	0	0	0	0	1	387
5	0	1	0	2	0	373	0	0	0	0	376
6	0	0	0	0	0	0	377	0	0	0	377
7	0	2	0	0	0	0	0	383	1	1	387
8	0	4	0	2	1	0	0	0	370	3	380
9	0	1	1	2	4	2	0	0	0	372	382
Total	373	399	376	395	392	376	377	383	372	382	3823

	Predicted Class										
True Class	0	1	2	3	4	5	6	7	8	9	Total
0	168	0	0	1	3	1	0	0	3	2	178
1	0	167	3	2	2	0	1	1	4	2	182
2	1	10	150	8	0	1	0	1	5	1	177
3	3	2	4	154	0	2	0	4	5	9	183
4	0	4	1	0	170	2	3	0	0	1	181
5	0	2	2	2	0	168	2	1	2	3	182
6	0	0	0	0	2	1	175	0	3	0	181
7	0	4	0	0	16	2	0	144	2	11	179
8	1	15	3	6	2	3	0	0	138	6	174
9	1	1	0	15	5	8	0	0	0	150	180
Total	174	205	163	188	200	188	181	179	151	185	1797

2.3 PRECISIONS

$$Prec_{training} = 0.989 \quad (2.3)$$

$$Prec_{training}(byclass) = [1.0, 0.972, 0.997, 0.977, 0.979, 0.992, 1.0, 1.0, 0.994, 0.978] \quad (2.4)$$

$$Prec_{test} = 0.885 \quad (2.5)$$

$$Prec_{test}(byclass) = [0.965, 0.814, 0.920, 0.819, 0.85, 0.893, 0.966, 0.953, 0.851, 0.810] \quad (2.6)$$

2.4 EXPLANATION OF THE CODE

The assignment has been written in **Python 3.6.2** with **Sublime Text 3.1.1** IDE. In order to run the assignment code, the following Python libraries are required;

- pandas
- numpy
- warnings
- sklearn

Simply run **question2.py**. You will see precisions, accuracies and confusion matrixes as a result.

Tuning algorithm used to get best accuracy. As a result, following hyper-parameters selected for the classifier to get best accuracy result:

- criterion = 'entropy'
- max_depth = 10
- random_state = 1337
- min_samples_split= 5

3 MULTILAYER PERCEPTRON CLASSIFICATION

3.1 ACCURACIES

$$Acc_{training} = 1 \quad (3.1)$$

$$Acc_{test} = 0.972 \quad (3.2)$$

3.2 CONFUSION MATRICES (TRAINING AND TEST SET, RESPECTIVELY)

	Predicted Class										
True Class	0	1	2	3	4	5	6	7	8	9	Total
0	376	0	0	0	0	0	0	0	0	0	376
1	0	389	0	0	0	0	0	0	0	0	389
2	0	0	380	0	0	0	0	0	0	0	380
3	0	0	0	389	0	0	0	0	0	0	389
4	0	0	0	0	387	0	0	0	0	0	387
5	0	0	0	0	0	376	0	0	0	0	376
6	0	0	0	0	0	0	377	0	0	0	377
7	0	0	0	0	0	0	0	387	0	0	387
8	0	0	0	0	0	0	0	0	380	0	380
9	0	0	0	0	0	0	0	0	0	382	382
Total	376	389	380	389	387	376	377	387	380	382	3823

	Predicted Class										
True Class	0	1	2	3	4	5	6	7	8	9	Total
0	178	0	0	0	0	0	0	0	0	0	178
1	0	180	0	0	0	0	0	0	2	0	182
2	0	1	176	0	0	0	0	0	0	0	177
3	0	0	2	175	0	3	0	0	2	1	183
4	0	1	0	0	179	0	0	0	0	1	181
5	0	0	0	1	1	179	0	0	0	1	182
6	0	0	0	0	1	0	178	0	1	0	181
7	0	1	0	0	0	5	0	168	0	5	179
8	0	7	0	0	0	0	0	0	160	7	174
9	0	0	0	2	1	2	0	0	1	174	180
Total	178	190	178	178	182	189	178	168	166	189	1797

3.3 PRECISIONS

$$Prec_{training} = 1 \quad (3.3)$$

$$Prec_{training}(byclass) = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0] \quad (3.4)$$

$$Prec_{test} = 0.973 \quad (3.5)$$

$$Prec_{test}(byclass) = [0.994, 0.947, 0.988, 0.983, 0.983, 0.947, 1.0, 1.0, 0.963, 0.920] \quad (3.6)$$

3.4 EXPLANATION OF THE CODE

The assignment has been written in **Python 3.6.2** with **Sublime Text 3.1.1** IDE. In order to run the assignment code, the following Python libraries are required;

- pandas
- numpy
- warnings
- sklearn

Simply run **question3.py**. You will see precisions, accuracies and confusion matrixes as a result.

Tuning algorithm used to get best accuracy. As a result, following hyper-parameters selected for the classifier to get best accuracy result:

- activation = 'relu'
- hidden_layer_sizes = 100
- random_state = 1337
- solver = 'adam'

4 SUPPORT VECTOR MACHINE CLASSIFICATION

4.1 ACCURACIES

$$Acc_{training} = 1 \quad (4.1)$$

$$Acc_{test} = 0.975 \quad (4.2)$$

4.2 CONFUSION MATRICES (TRAINING AND TEST SET, RESPECTIVELY)

True Class	Predicted Class										Total
	0	1	2	3	4	5	6	7	8	9	
0	376	0	0	0	0	0	0	0	0	0	376
1	0	389	0	0	0	0	0	0	0	0	389
2	0	0	380	0	0	0	0	0	0	0	380
3	0	0	0	389	0	0	0	0	0	0	389
4	0	0	0	0	387	0	0	0	0	0	387
5	0	0	0	0	0	376	0	0	0	0	376
6	0	0	0	0	0	0	377	0	0	0	377
7	0	0	0	0	0	0	0	387	0	0	387
8	0	0	0	0	0	0	0	0	380	0	380
9	0	0	0	0	0	0	0	0	0	382	382
Total	376	389	380	389	387	376	377	387	380	382	3823

True Class	Predicted Class										Total
	0	1	2	3	4	5	6	7	8	9	
0	178	0	0	0	0	0	0	0	0	0	178
1	0	180	0	0	0	0	1	0	1	0	182
2	0	2	175	0	0	0	0	0	0	0	177
3	0	0	1	175	0	2	0	2	2	1	183
4	0	0	0	0	179	0	0	0	1	1	181
5	0	0	0	0	0	181	0	0	0	1	182
6	0	0	0	0	0	0	181	0	0	0	181
7	0	0	0	0	0	5	0	168	0	6	179
8	0	5	0	1	0	1	0	0	163	4	174
9	1	0	0	4	0	1	0	0	2	172	180
Total	179	187	176	180	179	187	182	170	169	185	1797

4.3 PRECISIONS

$$Prec_{training} = 1 \quad (4.3)$$

$$Prec_{training}(byclass) = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0] \quad (4.4)$$

$$Prec_{test} = 0.975 \quad (4.5)$$

$$Prec_{test}(byclass) = [0.994, 0.962, 0.994, 0.972, 1.0, 0.952, 0.994, 0.988, 0.964, 0.929] \quad (4.6)$$

4.4 EXPLANATION OF THE CODE

The assignment has been written in **Python 3.6.2** with **Sublime Text 3.1.1** IDE. In order to run the assignment code, the following Python libraries are required;

- pandas
- numpy
- warnings
- sklearn

Simply run **question4.py**. You will see precisions, accuracies and confusion matrixes as a result.

Tuning algorithm used to get best accuracy. As a result, following hyper-parameters selected for the classifier to get best accuracy result:

- kernel = 'poly'
- decision_function_shape = 'ovo'
- random_state = 1337
- probability = True
- shrinking = True