James Bao (PSU ID: 934097519)

Machine Learning Assign #1

Using a 10-perceptron single layer network to classify handwritten digits

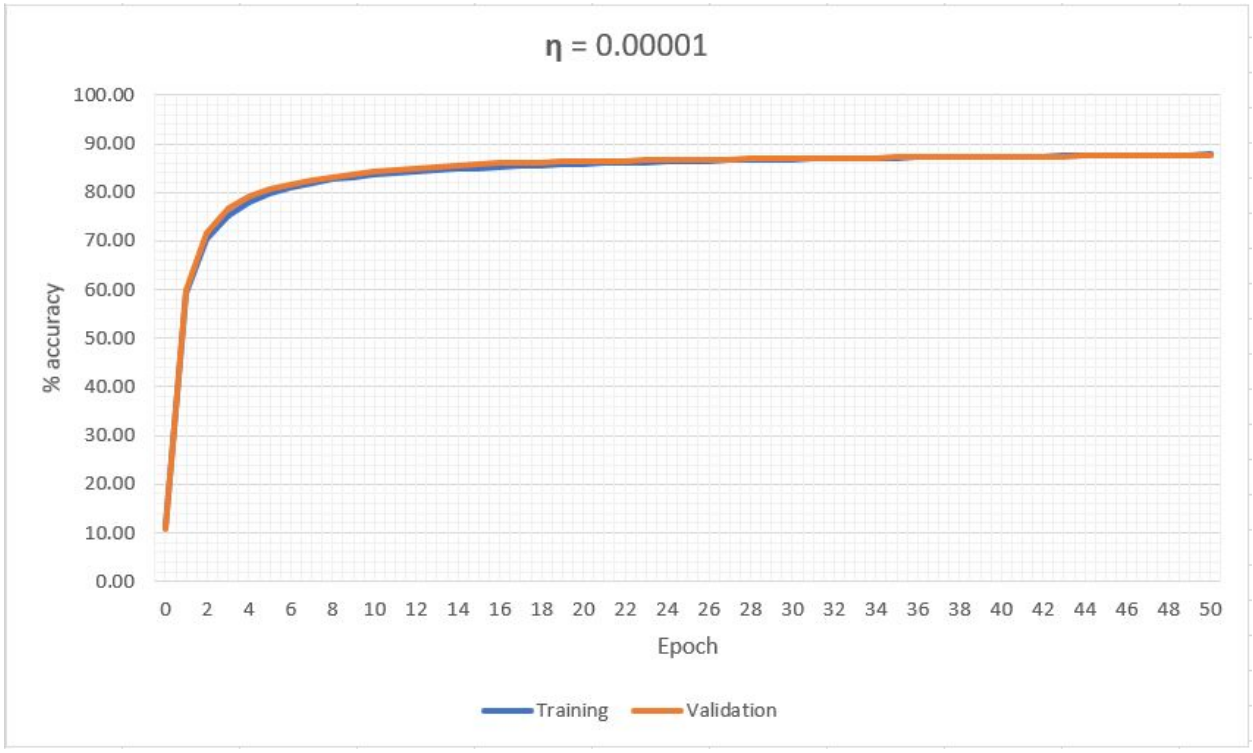
Introduction

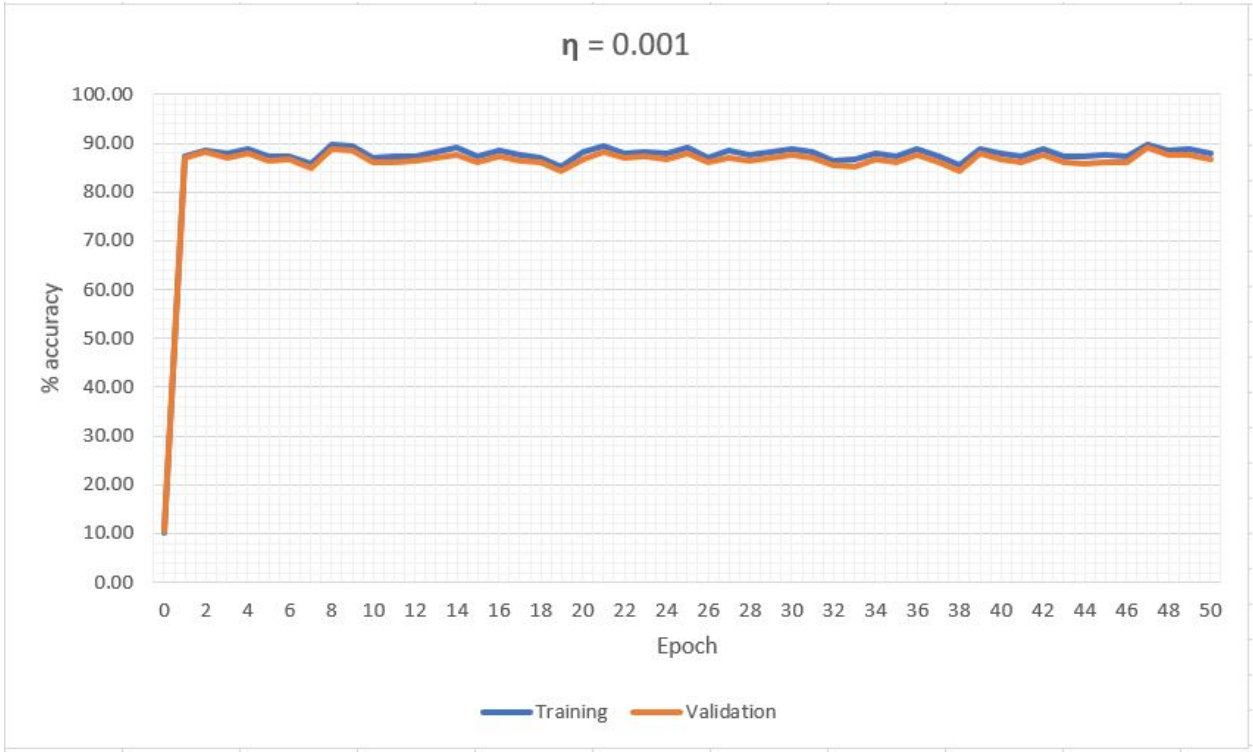
The purpose of this exercise was to design and implement a single-layer perceptron network of 10 perceptrons to classify handwritten digits pixel data stored in .csv format. The datasets used were a 60,000-sample training set and a 10,000-sample validation set; however smaller datasets were also used to test and debug the perceptron code. The perceptrons were trained at 3 different learning rates- 0.00001, 0.001, and 0.1. Weights were initialized at values between -0.05 and +0.05, and the training periods occurred over 50 epochs. The perceptron learning algorithm used stochastic gradient descent, which means that the weights on the inputs were updated after processing each sample in the training dataset, rather than after the sample was finished processing.

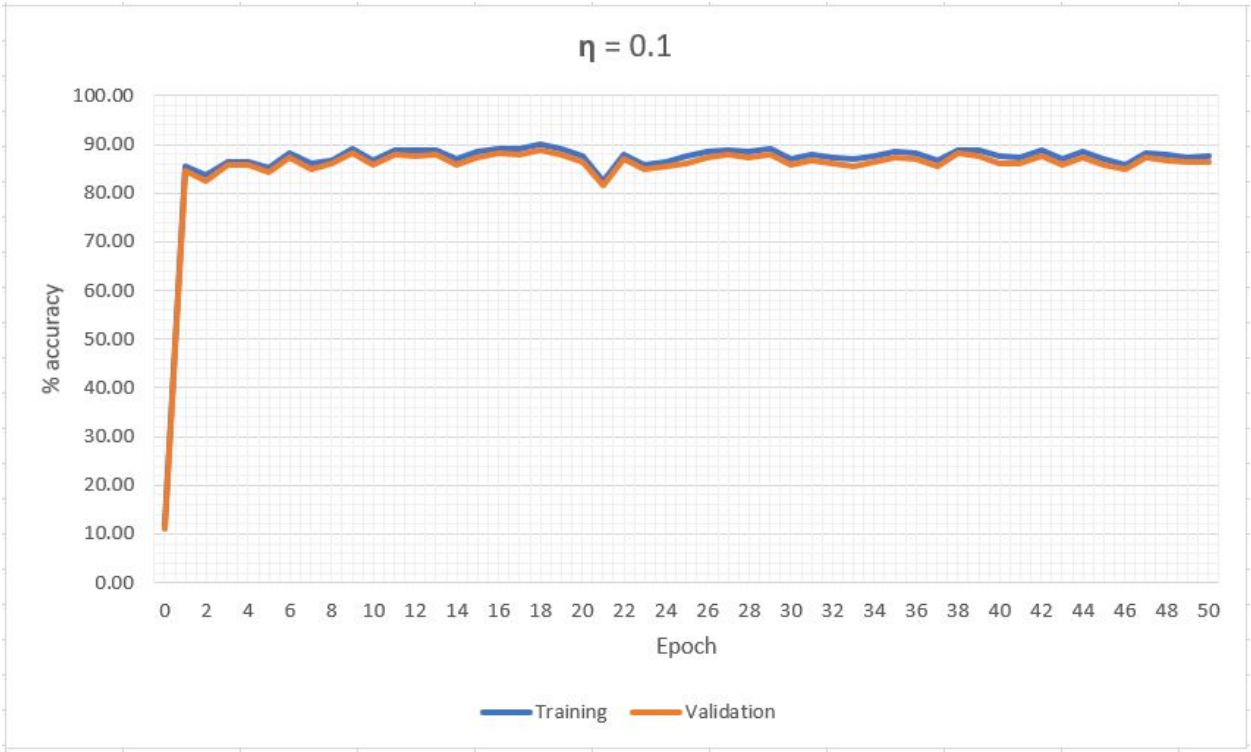
Accuracy plots

Each accuracy plot shows 2 curves, representing the training and validation datasets tested over 50 epochs. The algorithm code was edited in a way that required the training and validation accuracy data to be collected immediately after updating the weights during each epoch. Consequently, the training and validation accuracy curves exhibit the same upward and downward trends at all 3 learning rates. All 3 training sessions resulted in accuracy convergence around 87-88% for both training and validation data.

The 0.00001 learning rate run resulted in a slow, gradual increase in accuracy across the training period. While there were minor oscillations toward the end of the training period, they were less than 1% in magnitude and are not visible on the graph. By contrast, the 0.001 and 0.1 learning rate runs had faster accuracy improvements but larger oscillations.


η = 0.00001
100.00
90.00
80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
% accuracy
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
Epoch
Training
Validation


η = 0.001
100.00
90.00
80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
% accuracy
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
Epoch
Training
Validation



Confusion Matrices (from 10k sample validation dataset)

η = 0.00001 (actual = rows, expected = columns)

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	945	0	16	6	2	13	10	1	10	5
1	0	1087	9	3	3	2	3	8	13	8
2	2	4	863	21	12	5	7	21	9	3
3	6	5	32	879	7	57	3	7	38	20
4	2	1	10	3	879	21	14	9	11	56
5	9	4	13	45	4	711	16	7	60	15
6	8	3	16	3	6	21	887	2	14	2
7	2	0	12	14	4	10	1	913	9	43
8	5	29	51	28	19	40	17	8	788	39
9	1	2	10	8	46	12	0	52	22	818

η= 0.001 (actual = rows, expected = columns)

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	954	0	13	6	2	11	8	4	8	11
1	0	1110	38	1	6	4	4	9	20	8
2	0	0	738	3	2	1	3	10	2	0
3	5	4	79	938	5	122	3	19	34	37
4	2	0	7	2	847	10	2	5	5	20
5	3	1	5	6	5	584	9	0	10	12
6	8	5	41	5	25	25	921	1	16	2
7	2	1	15	15	9	17	1	960	11	85
8	5	14	91	31	40	114	7	7	865	74
9	1	0	5	3	41	4	0	13	3	760

η = 0.1 (actual = rows, expected = columns)

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	954	0	14	12	5	10	15	4	9	10
1	0	1119	61	6	5	4	5	13	18	8
2	0	3	748	5	6	0	4	6	2	0
3	1	1	30	834	4	30	2	5	6	7
4	2	0	7	0	830	4	2	1	4	13
5	11	1	22	47	9	728	28	1	27	29
6	2	3	21	3	8	12	878	1	5	0
7	4	1	20	24	20	17	1	977	14	99
8	6	7	106	73	63	84	23	9	885	165
9	0	0	3	6	32	3	0	11	4	678

Discussion of Results

While not necessarily clearly visible in the graphs, there were mild oscillations for the training dataset at all 3 learning rates. These occurred after the accuracy rate reached the upper 80s%, with larger oscillation magnitude and duration at the 0.1 learning rate, compared to the 0.001 learning rate. (The 0.00001 learning rate exhibited minimal oscillation, due to continuous gradual increases in the accuracy over time).

The confusion matrices demonstrate that the digits 0, 1, and 7 were easiest for the 10-perceptron network to classify- with >95% accuracy rates. 2, 5, and 9 on the other hand were the most difficult to classify; interestingly, all 3 of those digits were frequently mistaken for 8. There were no clear trends in which learning rate was best at classifying all 10 digits, since all three of them converged to the same accuracy rate. However, there was a trend of decreasing classification accuracy of 9's at higher learning rates.

The main reason the initial validation results were not accurate was due to a flawed program design. The original design involved making the set of target values an inherent component of the perceptron. Since the set of target values comes from a specific data set, this meant a perceptron object could only be used to classify one dataset, which therefore meant a separate perceptron needed to be created for classifying the validation dataset. This undermined the purpose of the experiment, since the validation dataset wouldn't be used to gauge the accuracy of the weights trained by the training dataset.