CptS 570: Assignment 1

Abrar Akhyer Abir

Question 1:

a. The decision boundary for voted perceptron is nonlinear. If we analyze the classification rule of voted perceptron, we can see that we take *sign* of $< w_i, x >$ for m weight vector and each time we get the sign, we multiply it with the confidence score. Finally, we will have same positive and negative values and after accumulating them whichever class, we will get that will be our result. It is not possible to represent such function with linear function. Hence, voted perceptron is not a linear function.

b. The decision boundary for averaged perceptron is linear. From the classification rule of average perceptron, we can see that we will finally get a single weight vector. Hence, it provides linear decision boundary.

Question 2:

For Linear update we have,

$$w_{t+1} = w_t + \tau y_t x_t \tag{1}$$

For Passive-Aggressive algorithm we try to achieve margin of *one*. So, if we force the constraint to hold as equality, we get:

$$1 = y_t((w_t + \tau y_t x_t) = y_t(w_t.x_t) + \tau||x_t||^2 \tag{2}$$

Now, if want the margin to be M, we will write:

$$M = y_t((w_t + \tau y_t x_t) = y_t(w_t.x_t) + \tau||x_t||^2 \tag{3}$$

After rearranging the equation, we get:

$$\tau = \frac{M - y_t(w_t.x_t)}{||x_t||^2} \tag{4}$$

Question 3:

a. Since we are given, the importance factor of each training example, we will try to adapt this information in our algorithm. We can define a loss function which will put greater importance whenever we make a mistake a training example that have high importance factor. In other word, we can integrate the h_i with the learning rate.

b. The significance of the training example might be suggested by learning for the same example numerous times in order to solve this learning problem utilizing the standard perceptron technique. We preprocess the data based on the importance factor, we can reproduce those data more times and those data will used to train a perceptron classifer which will be able to correctly predict those data.

Question 4:

a. Whenever perceptron made a mistake on positive examples which are only 10% of the dataset, we can increased the learning rate in order to specify the importance of positive examples correctly. This will emphasize on learning these positive examples more accurately.

b. The imbalanced data problem can be solved multiple ways. One of the solution is we can down sample the data. Since positive examples are lower here, we will take equal amount of negative examples as positive example and train using our standard perceptron. We will do this until we run out of training examples. Finally, we can take the weight vector that have the highest accuracy or we can take average of the weights to get the final classifer.

Question 5.1:

a. From the Figure 1, we can see as number of iteration increases, the number of mistakes from both perceptron and passive aggresive decreases. Although it fluctuates sometimes, it eventually decreases. We also see that number mistakes for PA is slightly higher than perceptron. It is caused by the conservative behavior of PA.

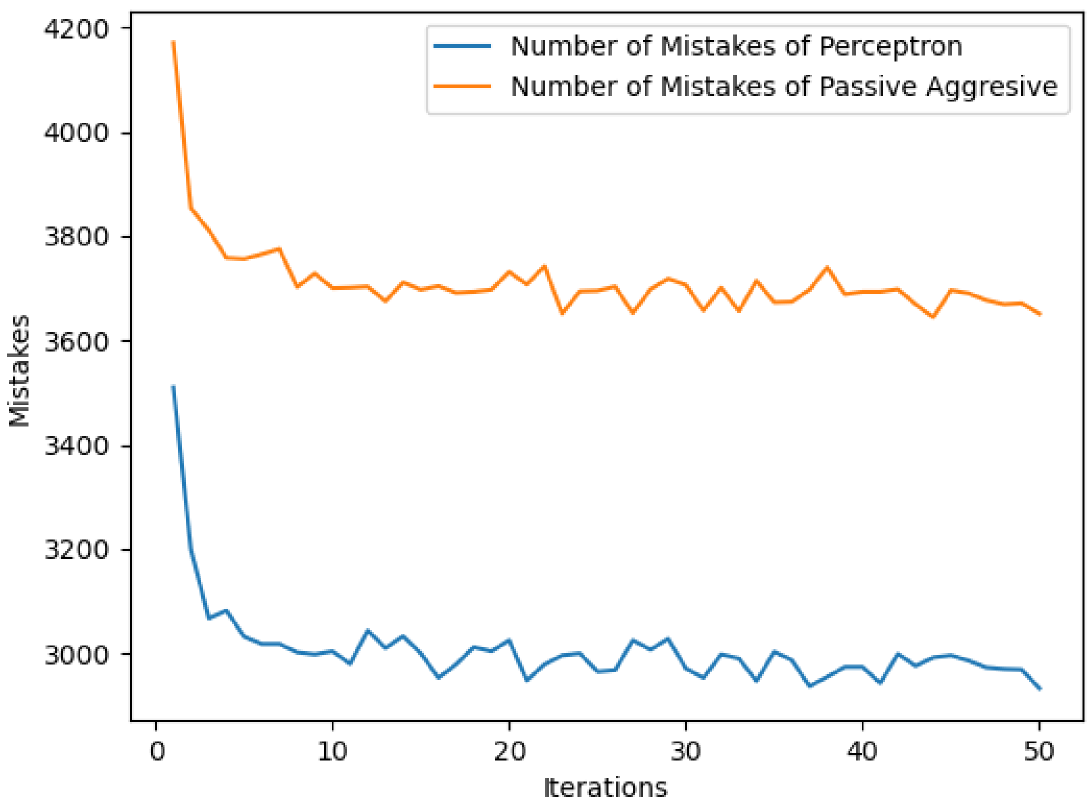

Figure 1. Number of mistake per iteration

b. From the Figure 2, in case of perceptron, we can see that it achieves relatively good accuracy both on training and testing data. We get highest training accuracy on early stage of the perceptron. Same as for testing accuracy. In second iteration, we get the highest accuracy among 20 iteration. In most of the cases, training accuracy is higher than testing accuracy.

In case of PA, we also observe similar behavior. The testing and training accuracy is higher on early stage. However, most of the time perceptron outperformed PA. PA has slightly lower training and testing accuracy than perceptron. One point to note that the testing accuracy is always lower than the training accuracy for PA.

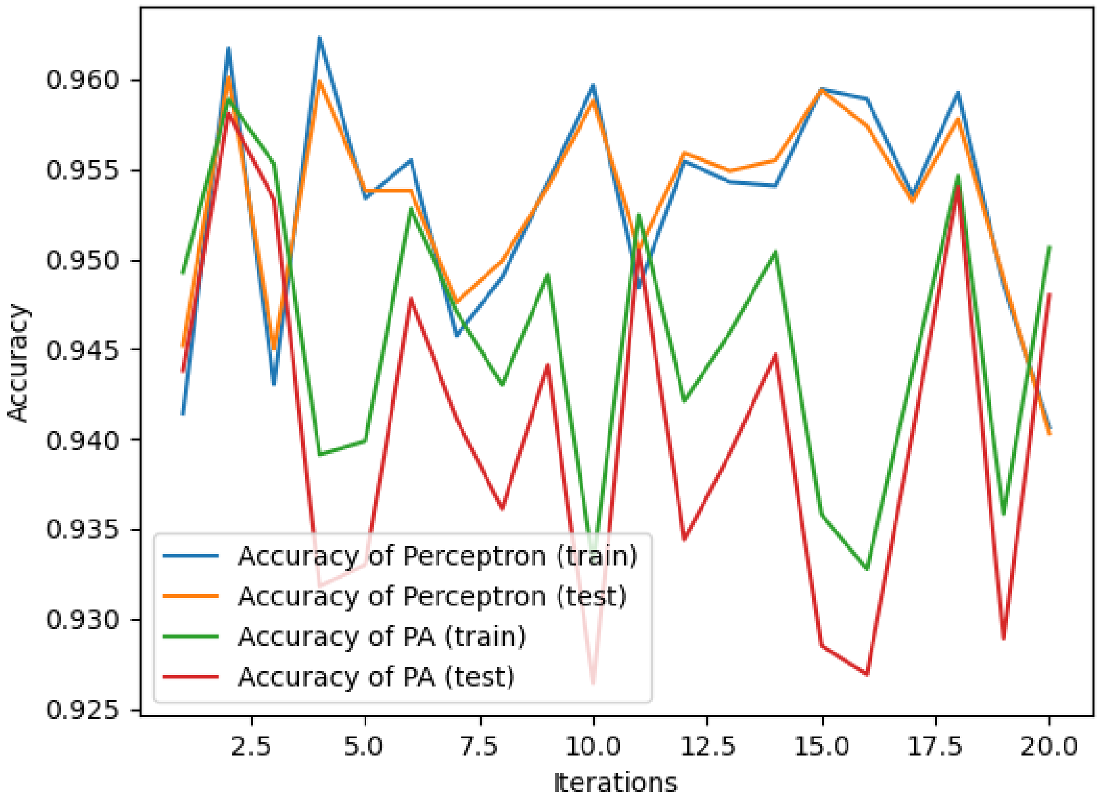

Figure 2. Training and testing accuracy for Perceptron and PA (Binary Classification)

c. From the Figure 3, in case of average perceptron, we can see that it always outperformed the plain perceptron. The training and testing accuracy also get increased after every iteration. However, the testing accuracy is lower than the training accuracy but still higher than standard perceptron.

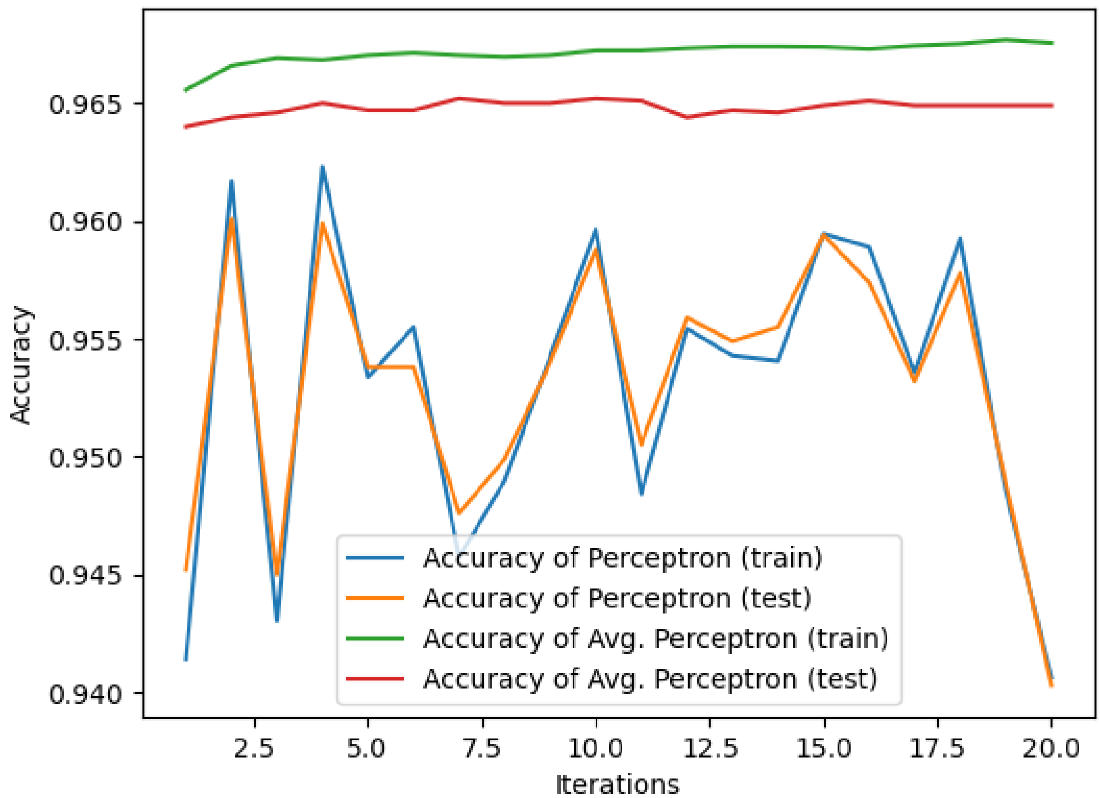

Figure 3. Training and testing accuracy for Perceptron and Average Perceptron (Binary Classification)

d. From the Figure 4, we can see that as the number of training data increases, the accuracy increased. In some cases of PA, the accuracy dropped drastically instead of having a large number of training data. Like other figures, we can also see that the accuracy of PA is slightly lower than perceptron.

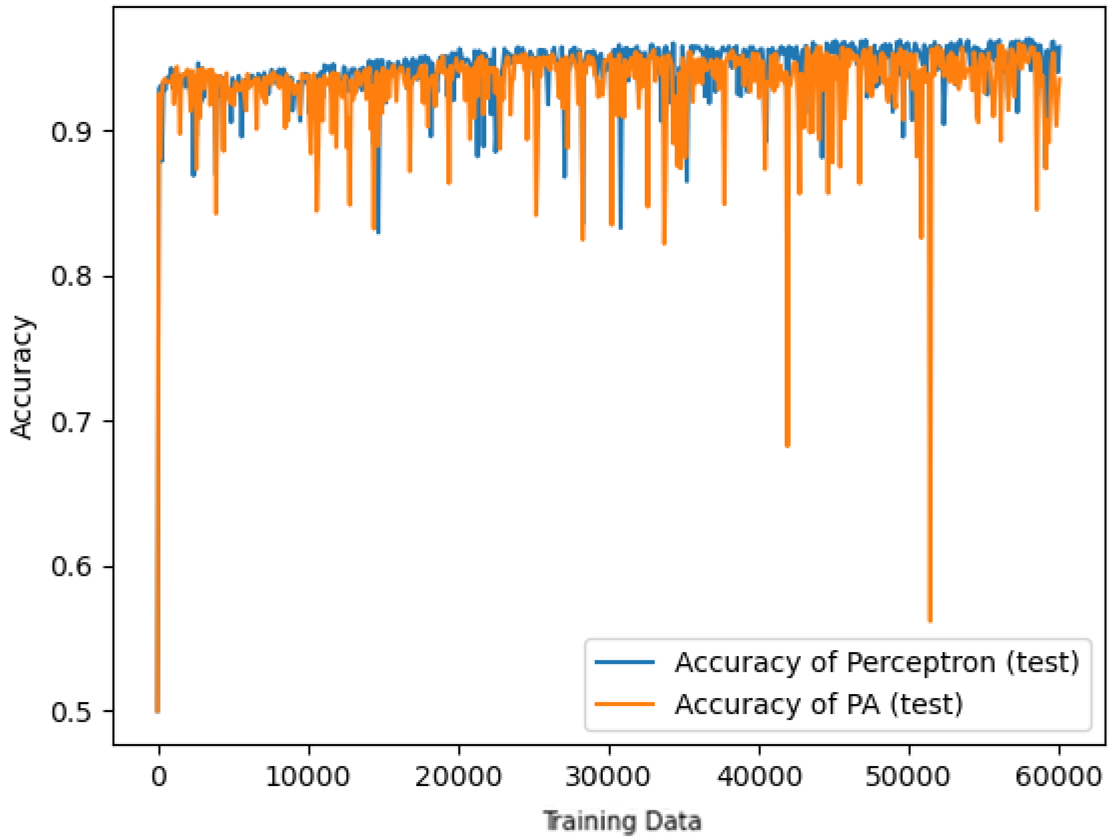

Figure 4. Learning curve for Perceptron and PA (Binary Classification)

Question 5.2:

a. From the Figure 5, we can see an interesting trend. The number of mistake for perceptron in case of multi-class classification is decreasing as the number of iteration increases. However, in case PA, it decreasing a little bit and again increased and it is not smooth. As a result, after 50 iteration, the number of mistakes of perceptron is much lower than PA.

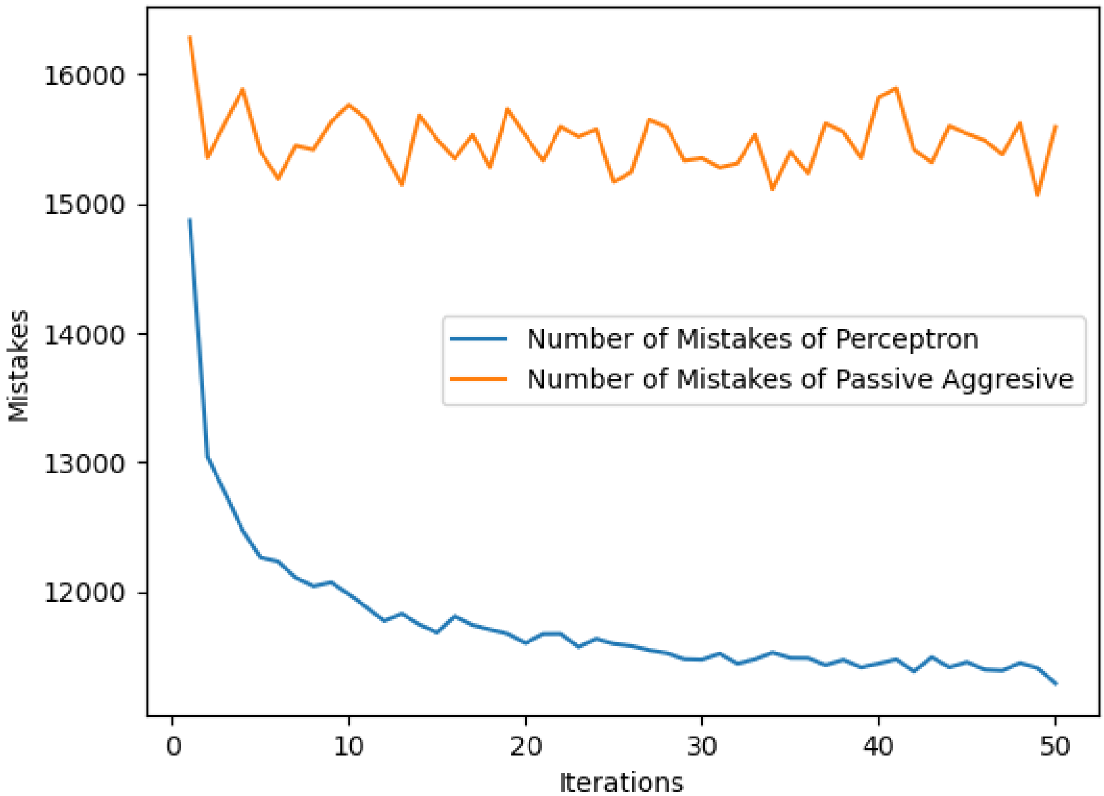

Figure 5. Number of mistake per iteration for Perceptron and PA (Multi-Class Classification)

b. From the Figure 6, we can see that the test and training accuracy for perceptron is higher in most of the cases than PA. Overall, the accuracy get declined than binary classification for both of the algorithms. Regarding the testing and training accuracy, testing accuracy is slightly lower than training accuracy in case of perceptron. For PA, they are almost same.

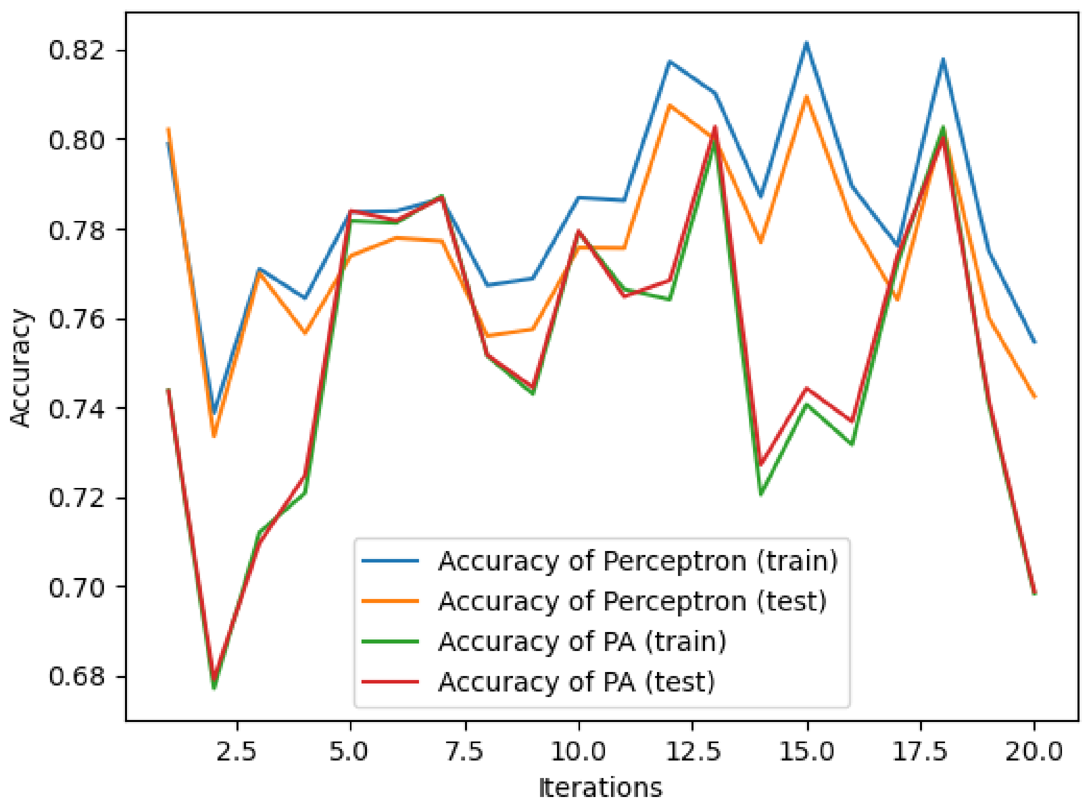

Figure 6. Training and testing accuracy for Perceptron and PA (Multi-Class Classification)

c. From the Figure 7, we can see that the average perceptron is also performing better than plain perceptron in multi class classification. However, the accuracy is lower than binary classification. We see a similar trend in training and testing accuracy like testing accuracy is lower than the training accuracy. The training accuracy increases after each iteration whereas the testing accuracy remain almost same for average perceptron.

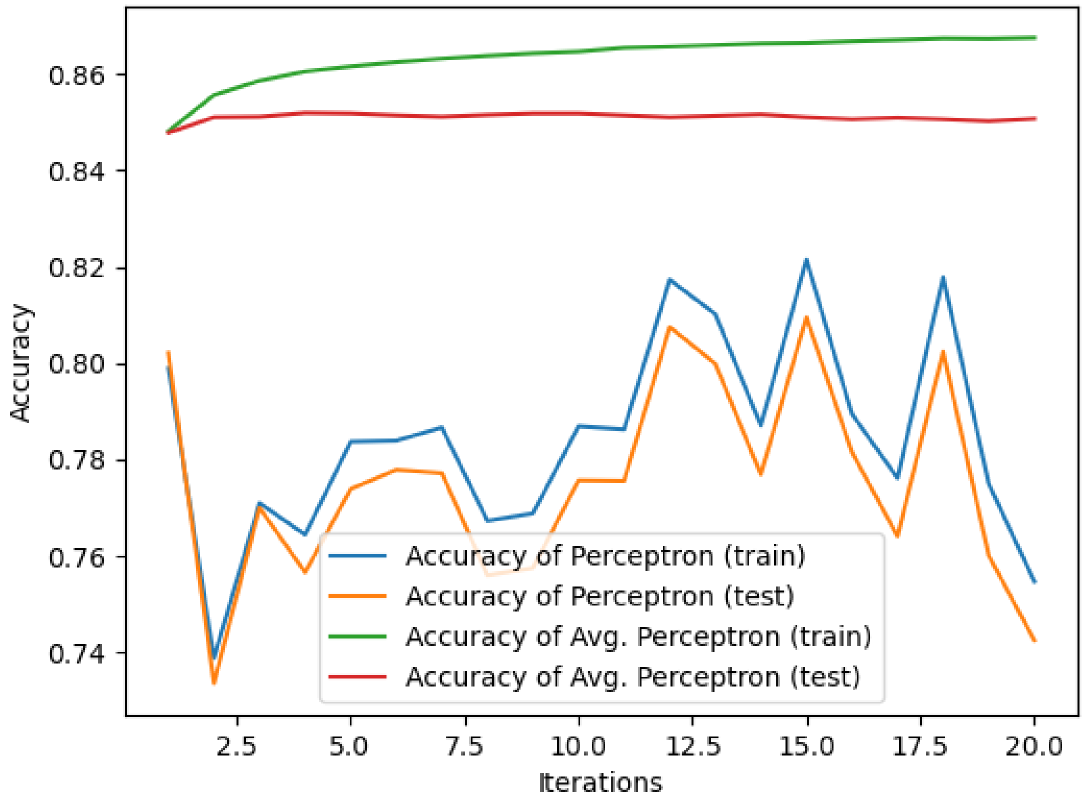

Figure 7. Training and testing accuracy for Perceptron and Average Perceptron (Multi-Class Classification)

d. From the Figure 8, we can see that as the data increases, the testing accuracy increases as well.

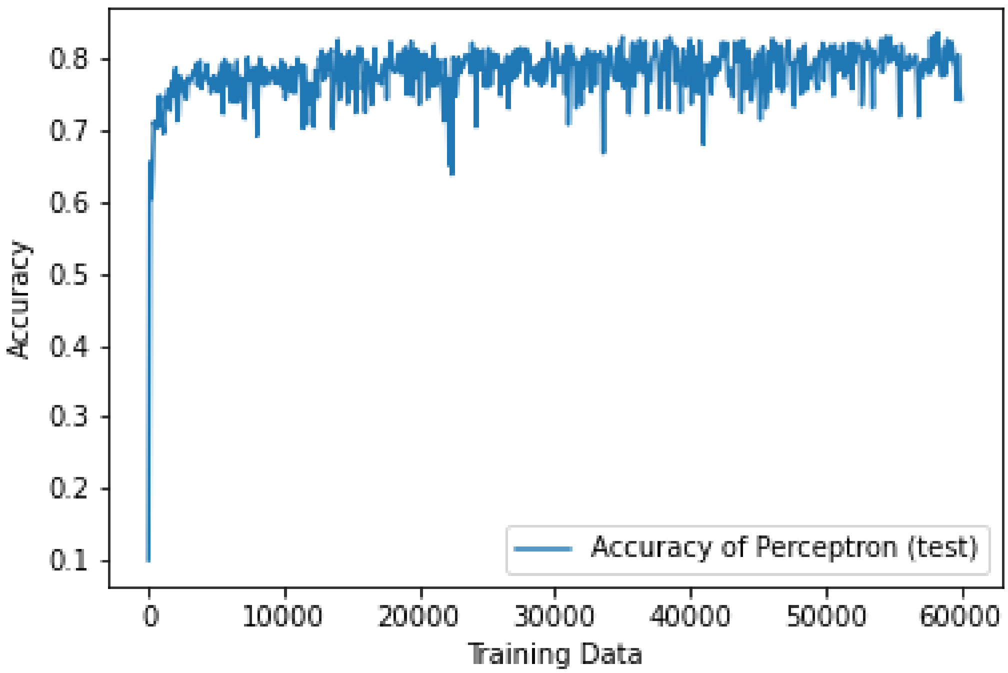

Figure 8. General learning curve for Perceptron (Multi-Class Classification)